
La
Odisea
del
Software
Libre
Compilador
Rodrigo
Gastón
Manresa

La Odisea del Software Libre



1.ª edición – 2.ª revisión:
Salta, Argentina - Julio de 2020
Título:
La Odisea del Software Libre
Coordinador:
Rodrigo Gastón Manresa
Código de Registro:
2007274870542



Edición:
Digital o Electrónica - Fénix Computación

Colaboraron en el presente libro (nombrados de acuerdo a aparición de sus capítulos):

- **Manresa, Rodrigo Gastón**
- **Suárez, Gonzalo Javier**
- **Rodríguez Zubieta, Mauro Esteban**
- **Salgado, Rogelio Augusto**
- **Díaz Pepenal, Elizabeth Carolina**
- **Rodó, Emilce Isabel**
- **Rodríguez, Natalia del Valle**
- **Arancibia, Paula Betiana**
- **Ventura, Jorge Luis**
- **López Pereyra, Marcos Alejandro**
- **Giliberti, Miguel Ángel**
- **Moya, Ángel Yael Eduardo**

¡copia y distribuí este libro!

Compartir no es delito

Prólogo:

En los presentes capítulos encontrarán la más variada información sobre el mundo del software libre, desde sus inicios, importancia y proyección, pasando por el software más destacado, hasta llegar a las actividades que se realizan en torno a esta innovadora iniciativa; durante su lectura no tratamos de convencer a los usuarios de mudarse de un software a otro; sino que tratamos de mostrar y dar a conocer las posibilidades, las múltiples aplicaciones-opciones que la mayoría desconoce y que aún no se animan a experimentar; pero como se destaca en este libro el tema de la libertad, podemos concluir en que el usuario es libre de conocer y elegir lo que mejor crea conveniente.

Rodrigo Gastón Manresa

Dedicado a las futuras generaciones, quienes seguirán este camino del Software Libre.

ÍNDICE

LA REVOLUCIÓN DEL SOFTWARE

Rodrigo Gastón Manresa

INTRODUCCIÓN AL SOFTWARE

Presentación del Libro

No es ajeno decir que actualmente existe una gran máquina (que es la computadora) que en nuestros tiempos nos permite tener al alcance varias herramientas poderosas como son los sistemas operativos y los programas que justamente son parte del software que en informática es a lo que definimos como la parte lógica de la computadora, es decir todo lo funcional.

La computadora se ha posicionado como una herramienta indispensable para el trabajo, la diversión, la educación y otros campos, en este sentido no puede escapar a ningún aspecto social, en otras palabras la informática esta abarcando y se esta expandiendo a todas las áreas o sectores, es por eso que vi la necesidad de editar este libro, donde se podrá observar un tema tan importante de

la informática como lo es el software libre y sus beneficios tanto en la actualidad como en un futuro muy cercano.

El presente libro tratará de abrir la mente de los lectores en el sentido de darse cuenta de que no solo existe el software propietario o privativo, que la mayoría de los usuarios están acostumbrado a ver, sino que hay algo que va más allá de eso, que está luchando contra ese monopolio, que ocupará un lugar destacado y eso es el ¨software libre¨.

También se enfrentará el desafío de dar respuesta de por qué se debe tener más en cuenta al software libre, cuáles son las principales características, los aspectos positivos que posee y las ventajas que nos presenta por sobre el software privativo, para que de esta manera se saque el máximo provecho de este aparato que actualmente, en el mundo en que vivimos, no puede pasar desapercibido y que avanza a pasos agigantados.

¿Qué es el Software Libre?

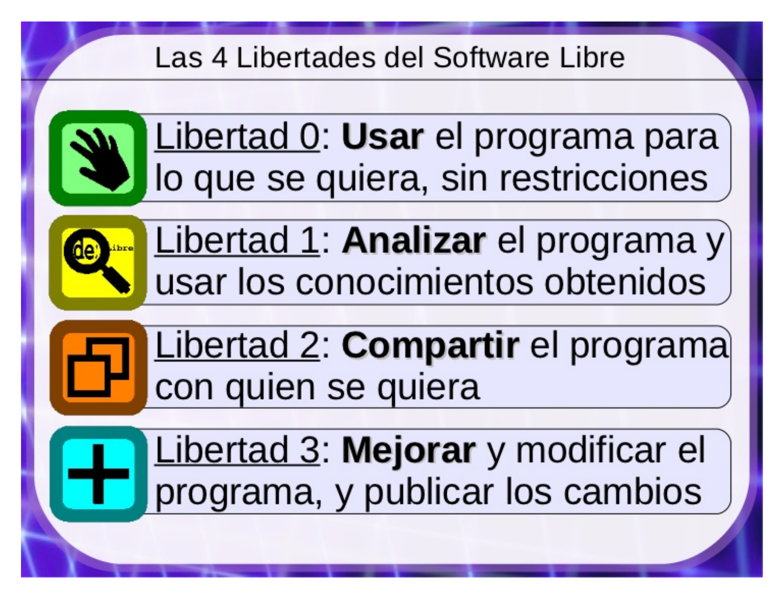
Es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos (por elección manifiesta de su autor), significa que los usuarios tienen la libertad de ejecutar, usar, copiar, distribuir, instalar, estudiar, modificar y mejorar el software. El software libre suele estar disponible gratuitamente, o al precio de costo; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar ¨software libre¨ a ¨software gratuito¨ (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente. En inglés la palabra ¨free¨ es ambigua ya que tiene dos significados que son ¨libre¨ y ¨gratis¨ es por eso que a veces se dice ¨libre software¨, en lugar de ¨free software¨, para mostrar que no se quiere decir que es gratuito, ya que como dije anteriormente no necesariamente el software libre es gratis.

Esta definición está asociada al nacimiento del movimiento de software libre, encabezado por **Richard Stallman** y la consecuente fundación en 1985 de la Free Software Foundation (FSF) que se dedica principalmente a difundir, promover o promocionar el desarrollo y uso del software libre, que coloca la libertad del usuario informático como un propósito ético fundamental.

La definición de software libre no contempla la cuestión del precio; un eslogan frecuentemente usado es "libre como en libertad, no como en cerveza gratis" o en inglés "Free as in freedom, not as in free beer" (aludiendo a la ambigüedad ya vista del término inglés "free"). Otros para evitar confusiones defienden el uso del término open source software (software de código abierto). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente

en los aspectos técnicos.

En concreto, el software libre implica que los usuarios tienen cuatro libertades esenciales:



* Para las libertades (1) y (3) el acceso al código fuente es una condición necesaria, por lo tanto el software debe ser de ¨código abierto¨.

Si analizamos las cuatro libertades podemos decir que la primera libertad sobre el ¨**uso**¨ se refiere a que podemos utilizar el software como deseemos y

con cualquier propósito, la segunda libertad que corresponde al ¨**estudio**¨ se refiere a que podemos analizar como funciona el software y adaptarlo a nuestras propias necesidades, la tercera libertad que tiene que ver con la ¨**distribución**¨ se refiere a que podemos copiar y compartir el software para ayudar a otros usuarios, y la cuarta libertad que trata sobre ¨**mejora**¨ nos dice que podemos modificar el software y hacer públicos estos cambios para que la comunidad se beneficie.

Un programa es software libre si otorga a los usuarios todas estas libertades de manera adecuada. De lo contrario no es libre.

Analogía para entender mejor

Haciendo una analogía podríamos decir que el software privativo es como comprar un automóvil y que el fabricante del mismo no te deje cambiarle de color, reemplazarle las llantas o hasta inclusive ni pegarle alguna calco. Esto seria algo loco de

pensar porque si yo adquirí un auto, lo pague y ya es mio, tengo el derecho a hacer lo que quiera con ese producto.

El software libre justamente lo que pretende es que los sistemas o programas que son intangibles o lógicos se comporten de la misma manera que un producto tangible o físico, es decir, que al adquirirlo pueda tener la libertad de hacer lo que yo desee con él, sin restricciones o condiciones.

Ventajas del Software Libre

En este apartado fundamentaré porque se debe priorizar el software libre por sobre el software propietario y para eso primeramente voy a definir ambas clases de software:

- El Software Privativo/Propietario es aquel que puedo usarlo pero bajo ciertas condiciones o términos, es decir que tiene restricciones que NO permiten ver su interior,

es decir su código fuente, NI mucho menos modificarlo o ampliarlo.

- El Software Libre es aquel con el cual puedo hacer lo que yo quiera ya que respeta la “libertad” de descargarlo, ejecutarlo, usarlo, copiarlo, estudiarlo, modificarlo y redistribuirlo libremente en las computadoras que desee o necesite.

Como podemos apreciar con solo leer la definición de ambas clases de software, claramente podemos deducir que el software libre nos brinda muchos beneficios por sobre el software privativo ya que no solo nos permite ver como esta creado internamente u observar su código fuente sino también nos permite modificarlo y sobre todo no hay restricciones en cuanto a la instalación porque podemos instalarlo en la cantidad de computadoras que deseemos o redistribuirlo en la cantidad de soportes que se nos ocurra. En otras palabras las

ventajas del software libre frente a los de licencia privativa son varias como por ejemplo podemos apreciar que evidentemente se ahorra dinero actuando de manera legal y para el caso de un pirata no tiene que estar parcheando ni crackeando el software evitando así numerosos errores y dolores de cabeza.

En cuanto al **costo** el software libre en la mayoría de los casos (podríamos decir que en un 99%) es gratuito y no hay que pagar un solo centavo a diferencia del software privativo que por lo general siempre tiene un excesivo precio.

El software privativo tiene muchas **condiciones o términos** que prácticamente no nos dejan nada de libertad a diferencia del software libre que como ya dije anteriormente ofrece la libertad de hacer prácticamente lo que el usuario desee en cuanto a ejecución, distribución, modificación, etc.

Con respecto a la **instalación** del software quizás

antes instalar un sistema operativo GNU/Linux era un poco complicado y solo para usuarios avanzados pero esto ya no es así e inclusive la instalación de programas por medio de los ¨gestores de software¨ se hizo mucho más sencilla de lo que era antes por comandos.

Si hablamos de **compatibilidad** en cuanto a hardware hoy en día ya no es un problema y existen drivers tanto privativos como libres con soporte de los fabricantes y de la comunidad respectivamente. En cuanto a la compatibilidad del software se puede migrar fácilmente e inclusive se pueden abrir o modificar cualquier tipo de formatos de archivo por ejemplo documentos de procesadores de texto privativos se pueden abrir y editar tranquilamente en un procesador de texto libre sin inconveniente alguno.

Si tratamos el tema de la **interfaz gráfica** podemos decir que en el software libre hay una gran variedad

de escritorios de usuarios a diferencia del software privativo en donde hay que conformarse con la interfaz que ya viene por defecto.

En cuanto a **programas** se puede decir que varios conocidos son privativos (software como los desarrollados por Adobe y Microsoft) que solo funcionan bajo entornos Windows a diferencia de los libres que son multiplataformas (hay versiones para Windows, MacOS y GNU/Linux). También se puede decir que existen programas libres de renombre e importantes (como los desarrollados por Mozilla y el proyecto GNU), además siempre existen versiones alternativas que son equivalentes a los privativos y con ellas se pueden hacer prácticamente lo mismo. Si bien son menos lo bueno es que junto al sistema operativo vienen pre-instalados un conjunto de programas plenamente útil ya desde un primer momento sin versiones de prueba y NO hay que estar buscándolos e instalándolos pos instalación del sistema operativo.

Si bien los **juegos** son programas creo que merecen una atención especial y dicho de ese modo aquí puedo llegar a decir que los “gamers” se encontrarían con una de las pocas desventajas del software libre ya que la mayoría de los últimos juegos salen de forma privativa y son pocos los libres aunque van en aumento y existen buenos títulos libres bastantes entretenidos y con excelentes gráficas que tranquilamente pueden hacerles sombra a varios de los privativos.

Si nos referimos a la **estabilidad**, en el software libre siempre fue muy buena y es raro que se produzcan errores en programas libres o se congele (clave) un sistema operativo libre ya que al estar el código fuente a la vista, muchas personas colaboran para mejorarlo y corregirlo, a diferencia del software propietario que es propenso a tener más fallas o errores que solo lo pueden corregir los programadores de la empresa que lo crea y esto hace que demoren más tiempo en ello.

En lo que tiene que ver con el **manejo y uso** podemos decir que la mayor parte del software privativo es muy fácil de manejar y esto se debe a que por lo general son más conocidos y más usados por el común de las personas pero el software libre ha avanzado tanto que manejarlo es igual de intuitivo y sencillo por lo que solo es cuestión de acostumbrarse y esto esta generando cada vez más adeptos (esto hablando de computadoras personales ya que si nos referimos a servidores, celulares inteligentes y tablets el software libre lidera en estos sectores).

Si nos centramos en el **rendimiento** el software libre le saca una gran diferencia al software privativo ya que siempre fueron rápidos al centrarse más en la funcionalidad y al administrar bien los recursos de la computadora o el sistema, por el contrario el software privativo en general prioriza el aspecto visual lo que lo hace tener un desempeño pobre o menor.

En lo que se refiere a la **seguridad** el software libre gana por lejos: primero porque los sistemas libres, como por ejemplo GNU/Linux, tienen otra administración y nivel de seguridad que es mucho más fuerte y estricto que los privativos; segundo porque existen más amenazas (como virus, gusanos o troyanos) para el software privativo y esos virus no afectan al software libre; tercero porque los hackers prefieren atacar más al software privativo ya que por el momento es el más usado y de esta manera se afectaría a una mayor cantidad de computadoras o sistemas; y por último porque el software libre tiene menos agujeros de seguridad ya que al tener código abierto también es muy fácil de detectar alguna falencia de seguridad que se corrigen rápidamente (es obvio que cuantos más ojos mejor) en contraste con los parches de seguridad del software privativo que demoran más tiempo en llegar a los usuarios ya que su detección y desarrollo por lo general es bastante lento.

Por último no debemos confundir Seguridad con **Privacidad**, y por su modelo de desarrollo solo el software libre puede asegurar esta última y garantizar gracias al acceso y revisión publica del código abierto que no existen agujeros o puertas traseras por donde se filtren nuestros datos y propicien esta recolección de información.

Para cerrar este primer capítulo del libro voy a decir que la pregunta a realizarse aquí, es la siguiente: ¿Cuál software debo elegir y me conviene usar? y para responder a esta interrogante diré que en definitiva la decisión quedará en las manos de cada uno de nosotros de acuerdo a nuestras necesidades, aunque esta claramente evidenciado que el software libre es lo que se viene y la mejor opción que existe.

*** Fuentes consultadas:**

- https://www.gnu.org/philosophy/free-sw.es.html
- https://www.gnu.org/philosophy/philosophy.html
- https://fsf.org/es/recursos/que-es-el-software-libre

LAS LICENCIAS DE LA LIBERTAD

Gonzalo Javier Suárez

Rodrigo Gastón Manresa

Software Libre o Free Software

Es un software disponible para cualquiera que desee utilizarlo, copiarlo y distribuirlo, ya sea en su forma original o con modificaciones. La posibilidad de modificaciones implica que el código fuente está disponible. Si un programa es libre, puede ser potencialmente incluido en un sistema operativo también libre. Es importante no confundir software libre con software gratis, porque la libertad asociada al software libre de copiar, modificar y redistribuir, no significa gratuidad. Existen programas gratuitos que no pueden ser modificados ni redistribuidos.



GPL

La **Licencia Pública General GNU** (GNU General Public License GPL) es la licencia que acompaña los paquetes distribuidos por el Proyecto GNU, más una gran variedad de software que incluye el núcleo del sistema operativo Linux. La formulación de GPL es tal que en vez de limitar la distribución del software que protege, llega hasta impedir que este software sea integrado en software propietario. La GPL se basa en la legislación internacional de copyright, lo que debe garantizar cobertura legal para el software licenciado con GPL.

Con esta licencia se puede instalar y usar un programa GPL , en un ordenador o varios, así como modificarlo y distribuirlo, para ello lo único a lo que estas obligado es a facilitar con el programa binario el código fuente, es decir si usas código licenciado en GPL, tu desarrollo final tiene que estar licenciado GPL.

LGPL

La **Licencia Pública General Reducida de GNU**, o más conocida por su nombre en ingles ***GNU Lesser General Public License,*** es una licencia creada por la (FSF) que garantiza la libertad de compartir y modificar el software cubierto por ella, asegurando que el software es libre para todos sus usuarios. Esta licencia se aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo estableciendo que su trabajo puede ser distribuido bajo los términos de esta. El "Programa", utilizado en lo subsecuente, se refiere a cualquier programa o trabajo original, y el "trabajo basado en el Programa" significa que el programa o cualquier trabajo derivado del mismo bajo la ley de derechos de autor.



AGPL

Una **Licencia Pública General de Affero** (Affero General Public License), se engloba dentro de las licencias destinadas a modificar el derecho de autor (de una obra o trabajo ya sea informático, musical, de arte o cualquier otro documento), también llamadas "licencias *copyleft***"**, derivadas de la Licencia Pública General de GNU, que ya hemos comentado. La **Affero GPL** es íntegramente una GNU GPL con una cláusula nueva que añade la obligación de distribuir el software si éste se ejecuta para ofrecer servicios a través de una red de ordenadores.



Licencias Estilo BSD

Las licencias BSD cubren las distribuciones de software de Berkeley Software Distribution, además

de otros programas. Éstas son licencias consideradas “permisivas”, ya que imponen pocas restricciones sobre la forma de uso, alteraciones y redistribución del software. El software puede ser vendido y no hay obligaciones de incluir el código fuente. Estas licencias garantizan el crédito a los autores del software pero no intentan garantizar que las modificaciones futuras permanezcan siendo software libre.



Licencia PSFL

La **Python Software Foundation License**, anteriormente **Python License**, es una licencia de software libre permisiva, al estilo de la licencia BSD, es decir, que cumple con los requisitos OSI para ser declarada licencia de software libre; además, es

compatible con la licencia GPL. A diferencia de la licencia GPL, y como la mayoría de licencias tipo BSD, la licencia PSFL no es una licencia copyleft, y permite modificaciones del código fuente, así como la creación de trabajos derivados, sin requerir que ni las modificaciones ni los trabajos derivados tengan que ser a su vez de código abierto. La licencia PSFL está dentro de las listas de licencias aprobadas tanto por la Free Software Foundation como por la Open Source Initiative.



Licencia MIT/X11

La licencia MIT es una de tantas licencias de software que ha empleado el Instituto Tecnológico de Massachusetts (MIT, Massachusetts Institute of

Technology) a lo largo de su historia, y quizás debería llamarse más correctamente licencia X11, ya que es la licencia que llevaba este software de muestra de la información de manera gráfica X Window System originario del MIT en los años 1980. Pero ya sea como MIT o X11, su texto es idéntico.

El texto de la licencia no tiene copyright, lo que permite su modificación. No obstante esto, puede no ser recomendable e incluso muchas veces dentro del movimiento del software de código abierto desaconsejan el uso de este texto para una licencia, a no ser que se indique que es una modificación, y no la versión original.

La licencia MIT es muy parecida a la licencia BSD en cuanto a efectos.

MPL y derivadas

La licencia pública de Mozilla (en inglés Mozilla Public License o MPL) es una licencia de código abierto y de software libre. Fue desarrollada originalmente por Netsacape Communications

Corporation (una división de la empresa **América Online**), y más tarde su control fue traspasado a la Fundación Mozilla.

La licencia MPL cumple completamente con la definición de software de código abierto de la Opens Source Initiative (OSI) y con las cuatro libertades del software libre enunciadas por la Free Software Foundation (FSF). Sin embargo la MPL deja abierto el camino a una posible reutilización no libre del software, si el usuario así lo desea, sin restringir la utilización del código ni el licenciamiento bajo la misma licencia. La MPL promueve eficazmente la colaboración evitando el efecto "viral" de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (Netscape Public License,

posteriormente llamdada MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU. En el espectro de las licencias de software libre se la puede considerar adyacente a la licencia estilo BSD, pero perfeccionada.



Licencia CDDL

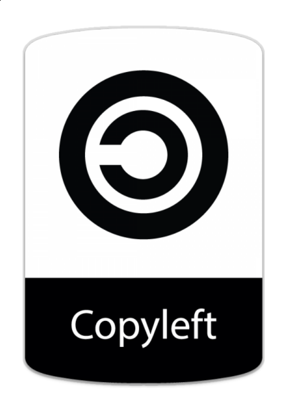
Common Development and Distribution License (Licencia Común de Desarrollo y Distribución), también conocida como **Sun Public License** (SPL)

versión 2, es una licencia de código abierto (OSI) y libre, producida por Sun Microsystems, basada en la Mozilla Public License o MPL, versión 1.1. La licencia CDDL fue enviada para su aprobación al Open Source Initiative el 1 de diciembre de 2004, y fue aprobada como una licencia de código abierto a mediados de enero de 2005. En el primer borrador hecho por el comité de divulgación de licencias OSI, la CDDL es una de las nueve licencias más populares, mundialmente usadas o con fuertes comunidades.

Copyleft

La mayoría de las licencias usadas en la publicación de software libre permite que los programas sean modificados y redistribuidos. Estas prácticas están generalmente prohibidas por la legislación internacional de copyright, que intenta impedir que alteraciones y copias sean efectuadas sin la autorización del o los autores. Las licencias que acompañan al software libre hacen uso de la legislación de copyright para impedir la utilización

no autorizada, pero estas licencias definen clara y explícitamente las condiciones bajo las cuales pueden realizarse copias, modificaciones y redistribuciones, con el fin de garantizar las libertades de modificar y redistribuir el software registrado. A esta versión de copyright, se le da el nombre de **copyleft**.



LPC

La **Licencia Pública Común** es una licencia publicada por **IBM** para su software. Esta tiene el objetivo de apoyar y fomentar el desarrollo colaborativo de software de código abierto al tiempo que conserva la capacidad de utilizar el contenido CPL con software licenciado bajo otras licencias,

incluyendo muchas licencias privativas. La CPL tiene algunos términos que se asemejan a los de la “GPL”, pero existen algunas diferencias claves. Para reducir el número de licencias de código abierto, IBM y la Fundación Eclipse, han acordado utilizar solamente la Eclipse Public License en el futuro. Open Source Initiative, por tanto, muestra la licencia pública común como obsoleta y sustituida por la EPL.

EPL

La **Licencia Pública Eclipse** (EPL) es una licencia utilizada por la Fundación Eclipse para su software. Sustituye a la Licencia Pública Común (CPL) y elimina ciertas condiciones relativas a los litigios sobre patentes. La Licencia Pública de Eclipse está diseñado para ser una licencia de software favorable a los negocios y cuenta con disposiciones más débiles que las licencias copyleft contemporáneas. El receptor de programas licenciados EPL pueden utilizar, modificar, copiar y distribuir el trabajo y las versiones modificadas, en

algunos casos están obligados a liberar sus propios cambios.



APSL

La **Apple Public Source License** es la licencia de código abierto y de software libre bajo la cual se publicó el sistema operativo de Apple, Darwin. Esta licencia fue adoptada de forma voluntaria para aumentar la colaboración de la comunidad de la que se origina gran parte de Darwin.

La primera versión de APSL fue aprobada por Open Source Initiative (OSI). La versión 2.0, lanzada el 29 de julio de 2003 es compatible con las directrices de la Free Software Foundation, y también está aprobada por la OSI. La Free Software Foundation aprobó la Apple Public Source License 2.0 como

una licencia libre ya que reúne los requisitos y afirma que está bien usar y mejorar el software ya publicado bajo esta licencia pero no recomienda utilizarla para proyectos nuevos porque es una licencia de software libre incompatible con la GPL de GNU y dado su copyleft parcial permitiría enlazar con archivos privativos. Por otro lado, la licencia requiere que si se lanzan productos derivados del código fuente original, el código debe ser accesible.

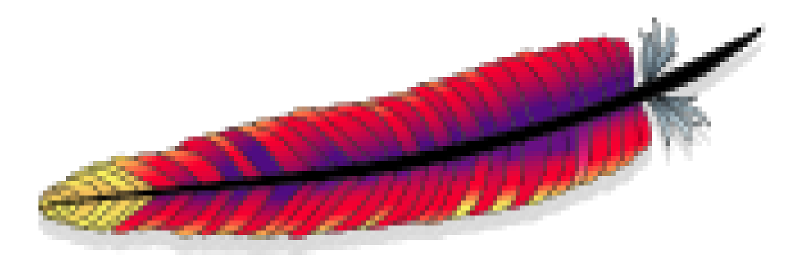
Apache License

La **licencia Apache** (**Apache Software License** para versiones anteriores a 2.0) es una licencia de software libre creada por la Apache Software Foundation (ASF). La licencia (con versiones 1.0, 1.1 y 2.0) requiere la conservación del aviso de copyright y el disclaimer, pero no es una licencia copyleft, ya que no requiere la redistribución del código fuente cuando se distribuyen versiones

modificadas ni siquiera que se tengan que distribuir como software libre/open source, sólo exige que se mantenga una noticia que informe a los receptores que en la distribución se ha usado código con la Licencia Apache.

Se deben añadir dos archivos en el directorio principal de los paquetes de software redistribuidos:

- LICENSE - Una copia de la licencia
- NOTICE - Un documento de texto, que incluye los "avisos" obligatorios del software presente en la distribución.



Licencia PHP

La **licencia PHP** es aquella bajo la cual se publica el lenguaje de programación PHP. De acuerdo a la

Free Software Foundation es una licencia de software libre no *copyleft* y una licencia de código abierto según la Open Source Initiative. Debido a la restricción en el uso del término "PHP", no es compatible con la licencia GPL.

La licencia PHP está diseñada para incentivar la distribución del código fuente. Se permite la redistribución del contenido licenciado en forma de código fuente o binaria siempre y cuando se cumplan los siguientes requisitos:

1. Se incluya la declaración de los derechos de autor de la licencia PHP;
2. La palabra "PHP" no se use en el título de las obras derivadas;
3. Se incluya el siguiente anuncio: “This product includes PHP software, freely available from <http://www.php.net/software/”.

Creative Commons

Las licencias **Creative Commons** o **CC** están inspiradas en la licencia GPL de la Free Software

Foundation, compartiendo buena parte de su filosofía. La idea principal detrás de ellas es posibilitar un modelo legal ayudado por herramientas informáticas, para así facilitar la distribución y el uso de contenidos.

No reemplazan a los derechos de autor, sino que se basan y se apoyan en estos para permitir modificar los términos y condiciones de acuerdo a la necesidad de cada obra. Sirven para llevar la postura extrema de *"Todos los derechos reservados"* hacia una más flexible, de *"Algunos derechos reservados"* o, en algunos casos, *"Sin derechos reservados"*. Estas licencias la pueden utilizar tanto personas como instituciones en casi cualquier obra creativa o de conocimiento.

Se debe tener en cuenta que fueron especialmente

diseñadas para contenidos, no para software, por lo que no es recomendable el uso de estas licencias para software libre o para documentación de software, y aunque nada impide hacerlo, lo mejor es utilizar licencias exclusivas para software (como las GPL de la Free Software Foundation, entre otras), que distinguen entre código fuente y código objeto, fundamental para el software. Para textos de soporte de un software en concreto existen también excelentes opciones recomendables y especificas como la **Free Documentation License** (Licencia de documentación libre) de la FSF.

Directrices Debian

Las directrices Debian es parte del contrato social realizado entre Debian y la comunidad de usuarios de software libre, y se denomina Debian Free Software Guidelines (DFSG). En esencia, este contrato contiene criterios para la distribución que incluyen, además de la exigencia de publicación del

código fuente:

1. La redistribución libre;
2. El código fuente debe ser incluido y debe poder ser redistribuido;
3. Todo trabajo derivado debe poder ser redistribuido bajo la misma licencia del original;
4. Puede haber restricciones en cuanto a la redistribución del código fuente, si el original fue modificado;
5. La licencia no puede discriminar a ninguna persona o grupo de personas, así como tampoco ninguna forma de utilización del software;
6. Los derechos otorgados no dependen del sitio en el que el software se encuentra;
7. La licencia no puede 'contaminar' a otro software.



*** Fuente consultada:** https://es.wikipedia.org/wiki/Licencia_de_software_libre

EL CAMINO DESDE SANTA CRUZ A EL CAPITÁN

Mauro Esteban Rodríguez Zubieta

EL CAMINO DESDE SANTA CRUZ A EL CAPITÁN: Los Sistemas libres no GNU

Si bien pareciese un recorrido desde la ciudad de nuestro vecino país de Bolivia a algún súper héroe de caricatura, en estas páginas trataremos de hacer una breve reseña a los sistemas operativos libres NO GNU que han dado origen y basamento a demás versiones UNIX y derivados de la normativa POSIX, la evolución de los mismos, entre otros.

1) UNIX – El padre de todos

Es un sistema operativo portable, multitarea y multiusuario; desarrollado, en principio, en 1969, por un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Dennis Ritchie, Ken Thompson y Douglas McIlroy.

El sistema, junto con todos los derechos fueron vendidos por AT&T a Novell, Inc. Esta vendió posteriormente el software a Santa Cruz Operation

en 1995, y esta, a su vez, lo revendió a Caldera Software en 2001, empresa que después se convirtió en el grupo SCO. Sin embargo, Novell siempre argumentó que solo vendió los derechos de uso del software, pero que retuvo el copyright sobre "UNIX®". En 2010, y tras una larga batalla legal, ésta ha pasado nuevamente a ser propiedad de Novell.



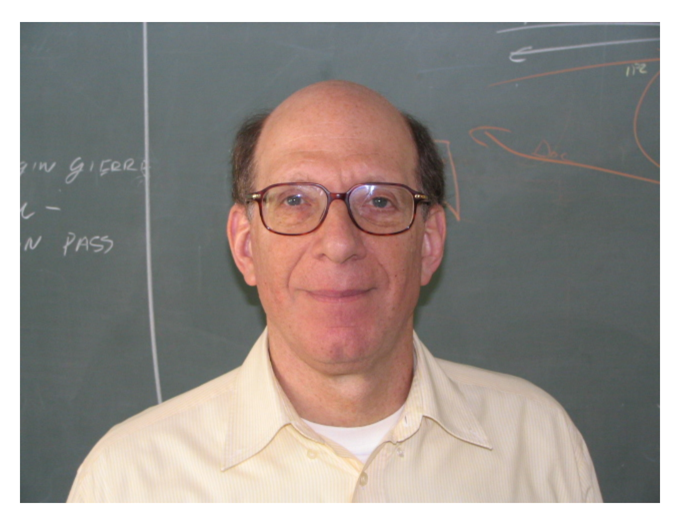
Unix – Sistema sin entorno gráfico

Se puede decir que cuando nace Unix lo hace como un software de código abierto pero no completamente libre; a partir de que fue vendido al

grupo Santa Cruz paso a ser privativo (de código cerrado) aunque de la versión de código abierto se desprendieron otros sistemas que siguieron los pasos de la libertad.

2) MINIX – El sistema operativo educacional

Este sistema es un clon del sistema operativo Unix distribuido junto con su código fuente y desarrollado por el profesor Andrew S. Tanenbaum en 1987.



Andrew Stuart Tanenbaum

Fue creado para enseñar a sus alumnos el diseño

de sistemas operativos en una universidad de Ámsterdam. La razón de su desarrollo fue porque Unix estaba bajo restricciones de licencia de AT&T, era demasiado complicado y corría sobre máquinas complejas; algo completamente antipedagógico.

Gracias a su reducido tamaño, diseño basado en el paradigma del micronúcleo, y su amplia documentación, resulta bastante apropiado para personas que desean instalar un sistema operativo compatible con Unix en su máquina personal así como aprender sobre su funcionamiento interno, ya que para las personas poco familiarizada con los elementos internos de un sistema operativo, MINIX es una buena opción que les permite entender casi todos los elementos del sistema con solo algunos meses de uso y estudio.

Debido al enfoque puramente educacional, el profesor Tanenbaum no dejaba que se modifique mucho el sistema ya que esto complicaría a los

alumnos para que entendiesen su materia en solo un semestre y fue así como un alumno llamado Linus Torvalds decidió empezar a escribir su propio núcleo compatible con UNIX en simbiosis con las herramientas del proyecto GNU y nació GNU/Linux del cual hablaremos en un capítulo aparte por ser el sistema emblema del software libre.

3) FreeDOS – Tras los pasos de MS-DOS

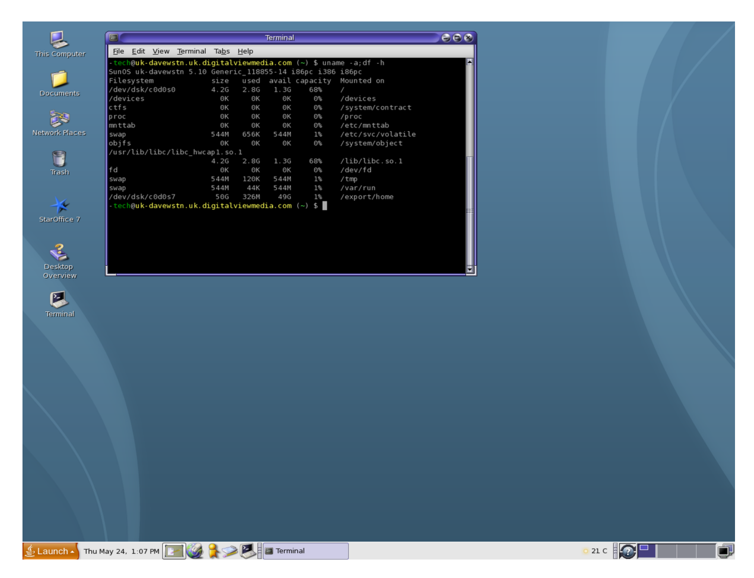
FreeDOS es un proyecto que aspira a crear un sistema operativo libre que sea totalmente compatible con las aplicaciones y los controladores de MS-DOS. El intérprete de línea de comandos usado por FreeDOS se llama FreeCOM.



Como principales diferencias con MS-DOS se puede decir que permite nombres de archivos largos, es de licencia libre (GPL) y tiene soportes para discos grandes.

4) De SunOS a Solaris y a Open Indiana

SUN Microsystems no podía quedar fuera de la jugada y en el año 1982 originalmente para la arquitectura M68k y basado en BSD dio a luz un sistema robusto que pronto se portabilizo a principios de los años 90 para la arquitectura X86 y también SPARC/UltraSPARC (procesadores propietarios de esta) y convirtiendo el nombre a un termino más comercial como Solaris.



Solaris 10.0 Usando JAVA DESKTOP SYSTEM

Es el sistema mejor adaptado para el soporte de SMP (multiprocesamiento simétrico) para servidores con gran cantidad de CPU.

Open Solaris: El código fuente de Solaris (con unas pocas excepciones) fue liberado bajo la licencia (CDDL) aunque es considerada una licencia de software libre, es incompatible con la GPL de Linux a principios del año 2000.



Login de Open Indiana X64

El Proyecto fue retomado luego que ORACLE al adquirir Sun dejo toda Royalty de Solaris como objetivo empresarial y se convirtió en Open Indiana el cual solo se edita para X86-X64.

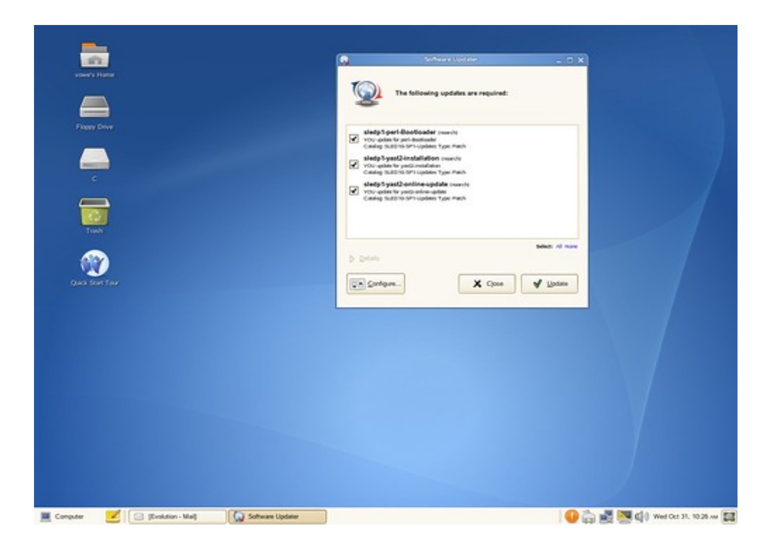
5) NOVELL – De liderar el mundo de las redes a ser un completo desconocido

Nadie olvidara a NOVELL por sus productos a mediados de los 80´ su Novell NetWare 4.12 estaba en el tope del mundo allá a principios de los años 90´ pero al igual que Nokia con su sistema operativo Symbian, Novell no pudo afrontar los tiempos venideros y sucumbió con su NetWare como un producto obsoleto.

Fue entonces que NOVELL adquirió la empresa Alemana SUSE Linux y transformo en dos Proyectos OpenSUSE y Novell Open Server, el cual nos dedicaremos a este último en estas líneas.

Novell Open Enterprise Server fue lanzado en marzo del 2005 y la arquitectura interna del producto consiste en una SUSE Linux Enterprise Server, a la cual se la ha añadido los demonios (procesos en segundo plano) que administran y controlan los servicios.

Aparte viene con añadidos, como es una para virtualización basada en el hypervisor Xen, soporte de 64 bits, soporte a un dominio de Windows, etc.



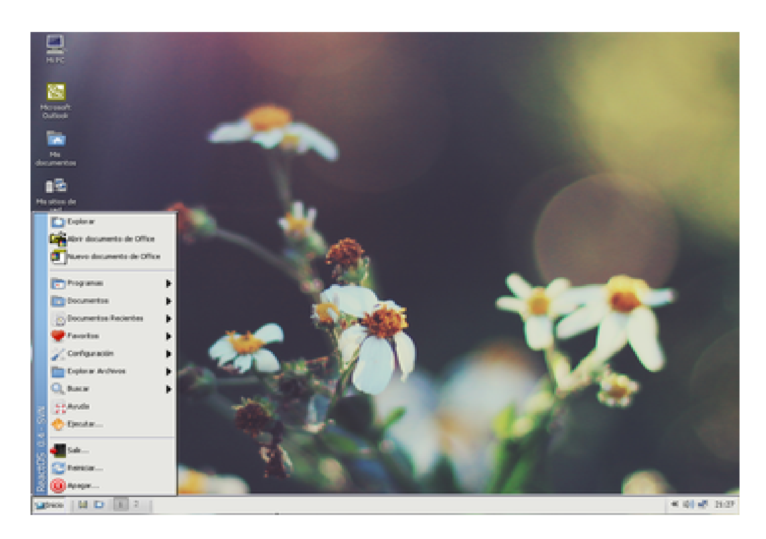
Escritorio de administración de Open Server 2 SP3

6) REACTOS – El proyecto interminable

Al fin un S.O. escrito totalmente en C y C++ y desde cero, originalmente planteado como un fork (desprendimiento) del proyecto llamado FreeWin95, en 1998 ReactOS se planteo como un sistema operativo compatible a nivel binario con Windows,

para permitir correr sus drivers y aplicaciones con una interfaz y entorno similar.

Considerado por Microsoft como una obra de arte de “Ingeniería Inversa” y denominándolo en ocasiones como los MILLI VANILLI de la informática, lejos de eso el proyecto ReactOS es un compendio de software con licencia BSD y GPL que en una etapa Beta procura poder ejecutar un porcentaje alto de programas de Windows sin tener que pagar una licencia a Microsoft.



Re Actos Explorer – Vista de Escritorio

7) BSD – La suma de todos los miedos



La Universidad de Berkeley (California) a principios de 1970 con la licencia UNIX para instituciones educativas que había brindado AT&T (Creador original de los sistemas POSIX/UNIX) ha sumado grandes contribuciones a la informática contemporánea:

- El manejo de memoria virtual paginado por demanda.
- El Control de Trabajos y Tareas.
- El Protocolo TCP/IP (Casi todas las derivaciones derivan del BSD 4.4).

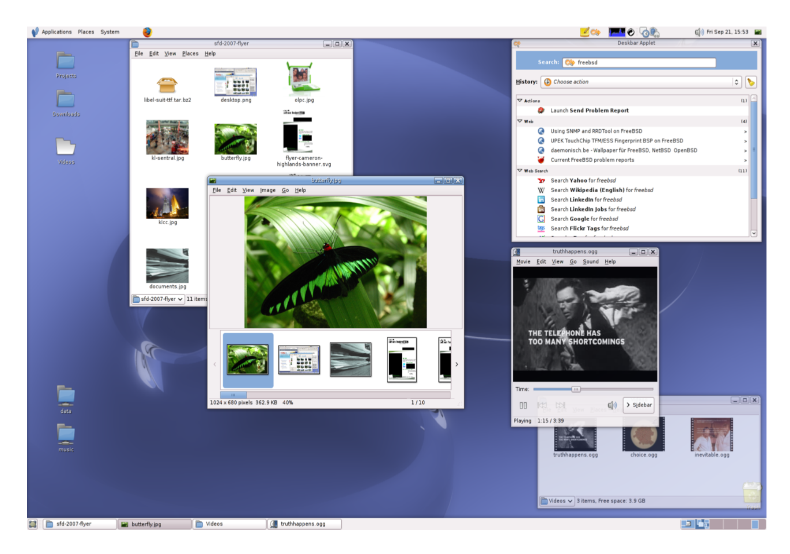
7.2) FREEBSD

FreeBSD está basado en la versión 4.4 BSDLite de UNIX, aunque no puede ser propiamente llamado UNIX al no haber adquirido la debida licencia de

The Open Group. Está disponible para plataformas X86, X64, MIPS, SPARC, PowerPC.

FreeBSD es compatible con binarios de varios sistemas operativos de tipo UNIX, incluyendo GNU/ LINUX, es asi que pueden ser ejecutadas aplicaciones desarrolladas para funcionar bajo el núcleo Linux.

La mascota del sistema operativo es el demonio llamado **Beastie** (BSD Daemon).



FreeBSD con escritorio GNOME

7.3) NETBSD

NetBSD es un sistema operativo de la familia Unix, de código abierto y libre que esta disponibles para muchísimas arquitecturas de hardware.

Su diseño y sus características avanzadas lo hacen ideal para multitud de aplicaciones. NetBSD ha surgido como resultado del esfuerzo de un gran número de personas que tienen como meta producir un sistema operativo tipo Unix accesible y libremente distribuible.

7.4) OPENBSD



OpenBSD es un sistema operativo libre tipo Unix multiplataforma. Es un descendiente de NetBSD, con un foco especial en la seguridad y la criptografía.

Este sistema operativo se concentra en la portabilidad, cumplimiento de normas y

regulaciones, corrección, seguridad proactiva y criptografía integrada. OpenBSD incluye emulación de binarios para la mayoría de los programas de los sistemas SVR4 (Solaris), FreeBSD, Linux, BSD/OS, SunOS y HP-UX. Se distribuye al igual que FreeBSD y NETBSD bajo la licencia BSD.

8) Darwin – De los felinos al spanglish

OS X, anteriormente llamado Mac OS X, es un entorno operativo basado en Unix, desarrollado y comercializado por Apple. Está basado en BSD, y se construyó sobre las tecnologías desarrolladas en NeXT entre mitad de los 80's y los 90's, cuando Apple adquirió esta compañía. Técnicamente, no es un sistema operativo, sino que incluye uno (Darwin, cuyo núcleo es XNU). Desde la versión 10.5 el sistema tiene Certificación UNIX, sigue fiel a su interfaz gráfica denominada AQUA y ha dejado de lado el nombre de felinos (Puma, Jaguar, Panther, Tiger, Leoppard, Lion, etc.) que se usaba en las

versiones anteriores (10.1 a 10.8) para pasar a nombres de lugares de California (donde se asienta la sede principal de Apple).



Mac OS X El Capitán

MacOS no es un software totalmente libre pero fue tendiendo a abrirse con el paso del tiempo ya que cada vez que sale una nueva versión posee más libertades y vendría a ser una especie de híbrido es decir un software con componentes cerrados pero también libres como Darwin (que es el **sistema operativo** que subyace dentro y esta bajo licencia APSL), Bonjour (licencia Apache), CUPS (licencia GPL), WebKit (licencia BSD) y otros tantos.

GNU/LINUX PARA TODOS LOS GUSTOS

Rodrigo Gastón Manresa

Mauro Esteban Rodríguez Zubieta

Rogelio Augusto Salgado

GNU/LINUX Y SUS DISTRIBUCIONES

Reseña histórica sobre GNU/Linux

El proyecto GNU se inició en 1.983 por **Richard Stallman** con el objetivo de desarrollar un sistema operativo libre compatible con UNIX. Para entender mejor, cabe recordar y destacar la anécdota de cuando Richard Stallman trabajaba en el MIT y a partir de un error que encontró en la impresora con la que trabajaba, quiso solucionarlo (ya que él era programador) y al ver que no tenia a disposición el código fuente, llamó al fabricante de la misma para solicitar que se lo den, pero este se lo negó y a partir de allí creó el GNU manifiesto. Definiendo por primera vez las 4 libertades básicas que hoy definen al Software Libre.

En 1.991 el ingeniero finlandés **Linus Torvalds** cuando estudiaba ciencias de la computación en la Universidad de Helsinki decidió comprar una

computadora generalmente para tener acceso vía módem a la red informática su Universidad, pero debido a que no le gustaba el sistema operativo Minix con el cual trabajaba, decidió empezar a crear uno similar compatible con UNIX, basándose en este empezó a crear un kernel para Minix y así fue que nació Linux que en un comienzo el propuso llamarlo Freax ya que pensaba que llamarlo Linux sería demasiado egocéntrico (pero se lo siguió conociendo como Linux, práctica que perdura hasta la actualidad).

Ya en el año 1.992 Torvalds liberó la licencia de Linux y de esta manera el proyecto GNU que ya había producido varias herramientas fundamentales para el manejo del sistema operativo, incluyendo varios programas, pero cuyo núcleo llamado “Hurd” aún no era lo suficientemente maduro y estable para usarse, adopto el núcleo Linux creado por Torvalds y se fusionaron creando un sistema operativo completo (es por esto que muchos

coinciden en que el termino más apropiado y correcto para llamarlo es GNU/Linux y no Linux a secas).[1]



Richard Stallman, iniciador de las herramientas GNU
(acrónimo recursivo que significa **G**nu **N**o es **U**nix)

Actualmente es desarrollado en colaboración por varios programadores y colaboradores de todo el mundo (una gran comunidad de software libre) además de contar con el apoyo de grandes empresas como IBM, Intel, HP, Dell, NVidia, AMD, entre otras tantas.

Supone el más grande proyecto colaborativo a nivel mundial en cuanto a software se refiere.

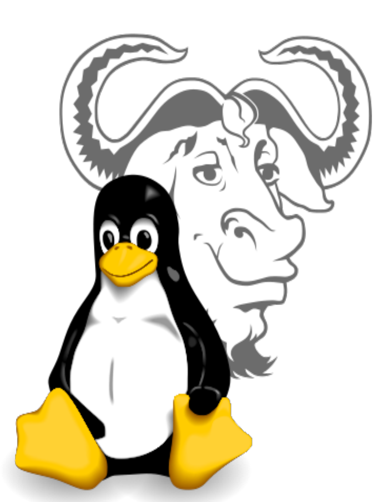
1 https://www.gnu.org/gnu/why-gnu-linux.html



Linus Torvalds, creador del núcleo Linux

Este sistema si bien no lidera el porcentaje de uso en computadoras de escritorio es muy usado en otros sectores como en la industria del cine, tanto es así, que el 95% de los servidores que usan los estudios de Hollywood para realizar películas de animación tienen instalada alguna distribución GNU/Linux. De hecho, Titanic, Avatar, El Hobbit, Buscando a Nemo, entre otras lo utilizaron para el desarrollo de sus efectos especiales. Además el 99% de las computadoras más potentes del mundo

utilizan GNU/Linux. Por ejemplo, los servidores de Google y el servidor que aloja la web de la Casa Blanca funcionan con GNU/Linux.



El ñu es el logo del proyecto GNU debido a que en inglés se refiere justamente a ese animal. El pingüino llamado Tux representa al núcleo Linux porque a Torvalds, de pequeño, le picó uno y le pareció gracioso que fuese la mascota.

Linux tiene millones de líneas de código y esta cantidad crece un 10% cada año. De estas millones de líneas, el kernel en sí supone un 5% y los drivers un 55%. Miles de líneas se agregan, se eliminan y se modifican cada día.

En el año 2002, Microsoft había acumulado un gasto de 421 millones dólares combatiendo la

expansión de Linux, según The Register, porque veía a este sistema como una amenaza y esto fue comprobado cuando Eric Raymond entro a los servidores y base de datos de Microsoft y publicó los “Documentos Halloween” que es el nombre con el que se conoce fuera de Microsoft a una serie de memorandos confidenciales sobre las estrategias potenciales de la compañía para desprestigiar al software libre, en particular a GNU/Linux.

GNU/Linux está presente en sectores muy delicados. Entre ellos se encuentran los trenes bala de Japón, el control del tráfico de San Francisco, la bolsa de Nueva York, el CERN, sistemas de control aéreo o el control de los reactores nucleares submarinos y buques de guerra nucleares.

Se implementa también en las computadoras de abordo de los vehículos, en la domótica de las casas inteligentes y en las computadoras de las principales agencias de investigación e inteligencia del mundo y estaciones espaciales como las de la

NASA y la Agencia Espacial Europea.

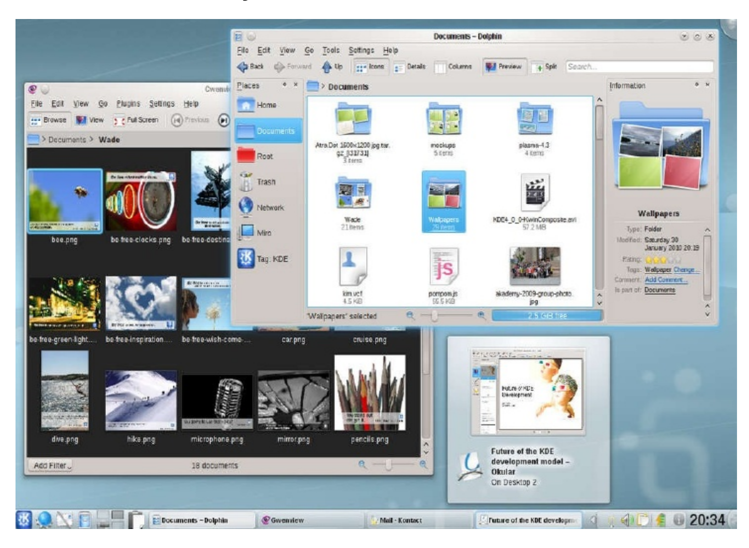
Muchos otros sistemas operativos usan el núcleo Linux como por ejemplo el archi conocido **Android** y otros que se usan en aparatos Smart.

Actualmente existen más de 300 distribuciones GNU/Linux activas alrededor del mundo y entre las más conocidas se destacan las siguientes: Slackware, Slax, Red Hat, Fedora, Mageia, CentOS, Debian, MX, Ubuntu, Mint, Elementary, Trisquel, OpenSuse, Arch, Manjaro y Gentoo.

1. Slackware Linux - Pioneros

Patrick Volkerding comenzó esta distribución que tiene el titulo de ser la más antigua (1991), fue basada originalmente para procesadores Intel 386 y a dado origen a decenas de distribuciones LINUX alternativas como College Linux, **SLAX**, Vector Linux, Zenwalk y otras siendo en el 2009 una de las últimas distribuciones en migrar a la arquitectura X64 (AMD – EMT64) y siendo una de las primeras y

pocas completas para procesadores ARM (Advanced Risc Machine) lo cual asegura a Slackware una larga vida al menos en dispositivos móviles con procesadores ARM-CORTEX-SNAPDRAGON y similares.



2. Red Hat Linux – El sombrero rojo

Siguiendo a Slackware RED HAT Inc en 1994 fue la primera en tener un sistema de instalación con coherencia de dependencia (RPM) y es la responsable de la creación y mantenimiento de Red

Hat Enterprice Linux (RHEL) que ha dado origen a otras familias de Linux como **Yellow Dog**, Mandrake Linux (llamado posteriormente Mandriva al fusionarse con Conectiva y luego **Mageia** al retomarse el proyecto por unos aficionados y ex-empleados), **Fedora** (que cuenta con el patrocinio y el respaldo de Red Hat pero es su versión no comercial), y otros como **CentOS** que es una distribución mantenida por la comunidad y derivada de los paquetes fuentes liberados por Red Hat, en otras palabras un clon “empresarial” pero gratuito.

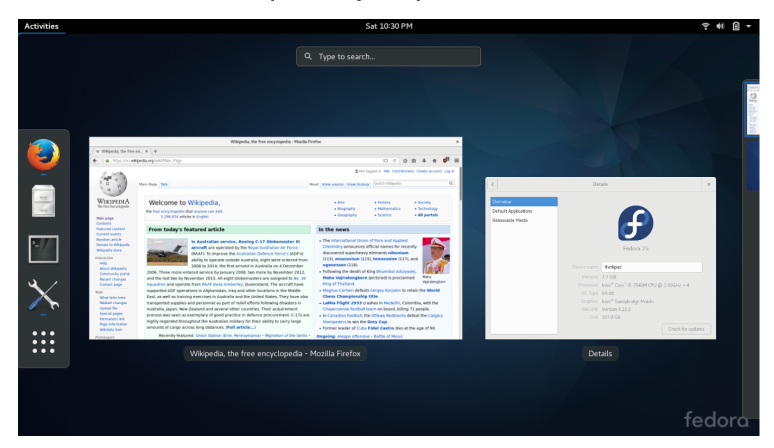


Red Hat fue vendido principalmente como sistema operativo en el área de servidores.

3. Fedora – La versión no comercial

Es una distribución GNU/Linux para propósitos generales basada en RPM, que se caracteriza por

ser un sistema estable, la cual es mantenida gracias a una comunidad internacional de ingenieros, diseñadores gráficos y usuarios que informan de fallos y prueban nuevas tecnologías. Cuenta con el respaldo y la promoción de Red Hat.



El proyecto no busca solo incluir software libre y de código abierto, sino ser el líder en ese ámbito tecnológico. Algo que hay que destacar es que los desarrolladores de Fedora prefieren hacer cambios en las fuentes originales en lugar de aplicar los parches específicos en su distribución, de esta forma se asegura que las actualizaciones estén disponibles para todas las variantes de GNU/Linux.

"Hablar de Fedora es hablar del rápido progreso del software libre y de código abierto."

4. Debian – Un Linux hasta en la sopa

Iniciado en 1992, es considerado el origen de la mayoría de las distribuciones ya que son muchas las que se basan en ella como las famosas **MX Linux** y **Ubuntu** o las de Seguridad **Kali y Parrot.** Es además la distribución que esta portada a más de 10 arquitecturas de procesadores.



Todas deben cumplir el contrato social de Debian:

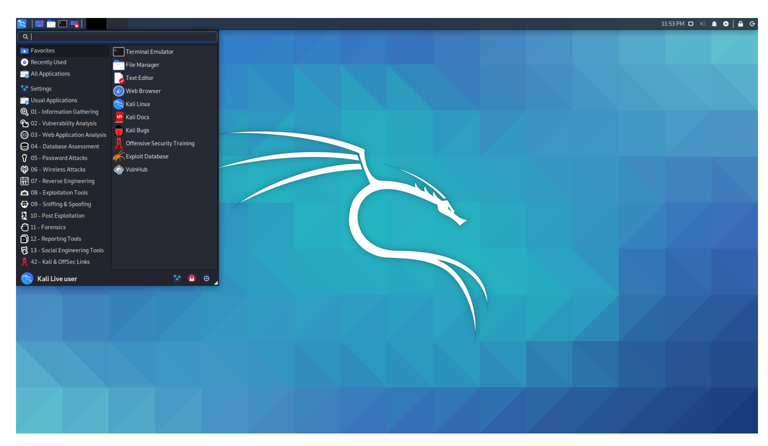
Debian sostiene que los fabricantes de un sistema operativo libre deberían proveer cuando un usuario les confía el control de una computadora ciertas garantías que incluyen:

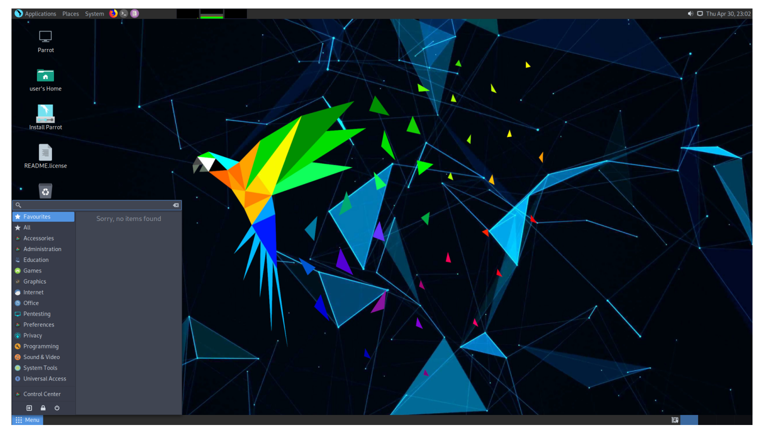
- Asegurarse de que el sistema operativo permanezca abierto y libre.
- Devolver las mejoras a la comunidad que hizo el sistema operativo posible.
- No esconder problemas en el software o la organización.
- Permanecer enfocado en los usuarios y el software que comenzó el fenómeno.
- Hacer el software posible de usar en problemas de uso cotidiano, aunque signifique utilizar software no libre.

5. Kali y Parrot – Seguridad y forense

Kali Linux (previamente conocido como BackTrack) y SO Parrot son dos distribuciones orientadas a la seguridad informática basadas en Debian con una

colección de herramientas diseñadas para las pruebas de penetración, evaluación y análisis de vulnerabilidades, auditoría, computación forense, ingeniería reversa, jaqueo, privacidad, anonimato y criptografía.





6. MX Linux – Priorizando el rendimiento

Distribución de GNU/Linux orientada a escritorios, basada en la rama estable de Debian, es un proyecto cooperativo entre las comunidades de **antiX** y el antiguo MEPIS, con el objetivo de utilizar las mejores herramientas y talentos de cada una de estas distribuciones.



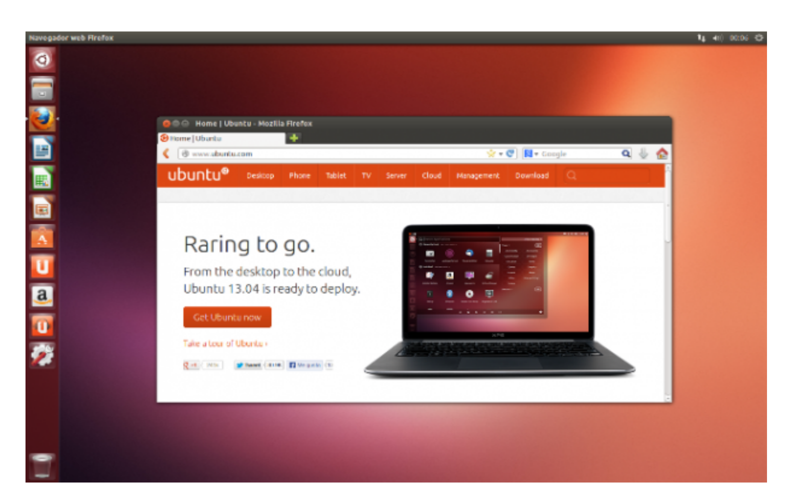
Usando Xfce como el escritorio por defecto, es un sistema operativo de medio peso diseñado para combinar un escritorio elegante y eficiente con una configuración simple, alta estabilidad, rendimiento sólido y un espacio de tamaño medio.

7. Ubuntu – La ética antes que todo

Su nombre proviene de la ÉTICA HOMÓNIMA, en la que se habla de la existencia de uno mismo como cooperación de los demás. La palabra Ubuntu es de origen africano y significa: "humanidad hacia otros", también "yo soy porque nosotros somos".

Desde hace tiempo Ubuntu es uno de los más usados en todo el mundo, lo que lo hace muy especial y único es que forma parte del proyecto software libre que destaca la libertad de los usuarios y la facilidad de uso e instalación. Su filosofía es: “GNU/Linux para seres humanos”, el objetivo principal es hacer un sistema operativo aún más accesible y fácil para los usuarios.

Dentro del mundo Linux su “share” del mercado es de aproximadamente el 60% lo cual indica la preferencia de los usuarios a esta distribución de la empresa Sudafricana-Británica (Canonical fundada por Mark Shuttleworth).

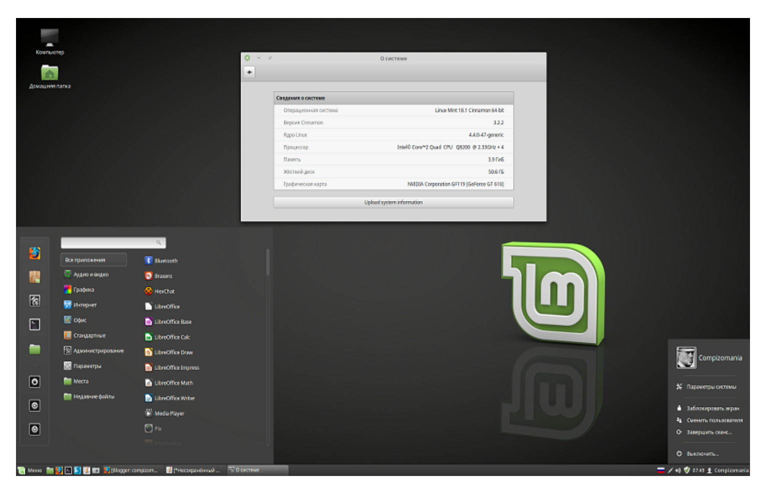


Cada seis meses se publica una nueva versión de Ubuntu. Esta recibe soporte por parte de Canonical durante nueve meses por medio de actualizaciones de seguridad, parches para bugs críticos y actualizaciones menores de programas. Las versiones LTS (*Long Term Support*), que se liberan cada dos años, reciben soporte durante cinco años en los Sistemas Escritorio.

8. Mint – Una versión fresca

Apareció en el año 2010 y sorprendió por ser una distribución basada en Ubuntu (que a su vez esta

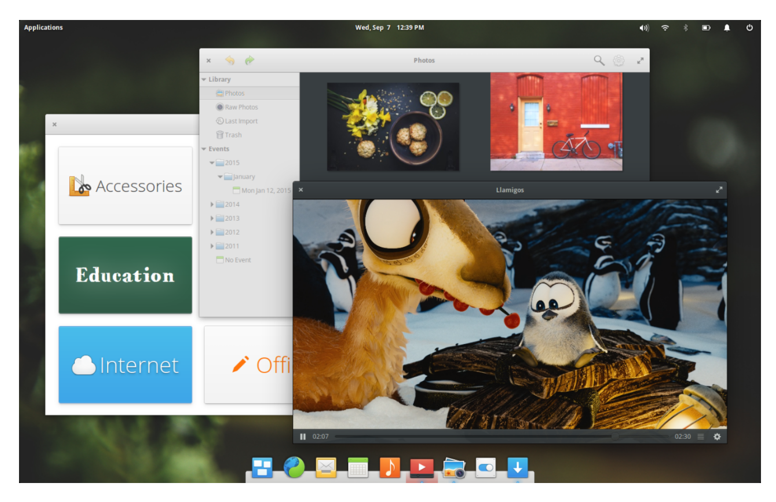
basada en Debian) fácil de instalar y fácil de usar.
Se aparto de los escritorios clásicos KDE y GNOME utilizando un nuevo entorno gráfico sobre X-Windows denominados Cinnamon y MATE (ambos tratando de proveer un entorno sencillo).



Para actualizaciones de sistema utiliza un sistema similar al .torrent pero con extensión .mint algo muy interesante. Incluye varios de los programas más usados, integra códecs de audio y vídeo multimedia y posee restricción a sitios web para adultos (Mint Nanny) entre otras soluciones Mint.
Es Lindo, es fácil, es sencillo es MINT.

9. Elementary – Simplicidad y elegancia

Elementary OS es una distribución basada en Ubuntu LTS. Usa un entorno de escritorio basado en GNOME con un *shell* propio llamado *Pantheon*. Dicho entorno destaca por ser más ligero y por la integración con otras aplicaciones de elementary OS como Plank (dock), Epiphany (el navegador web), Scratch (editor simple de texto) o Birdie (cliente de Twitter). Como gestor de ventanas usa Gala, basado en Mutter.



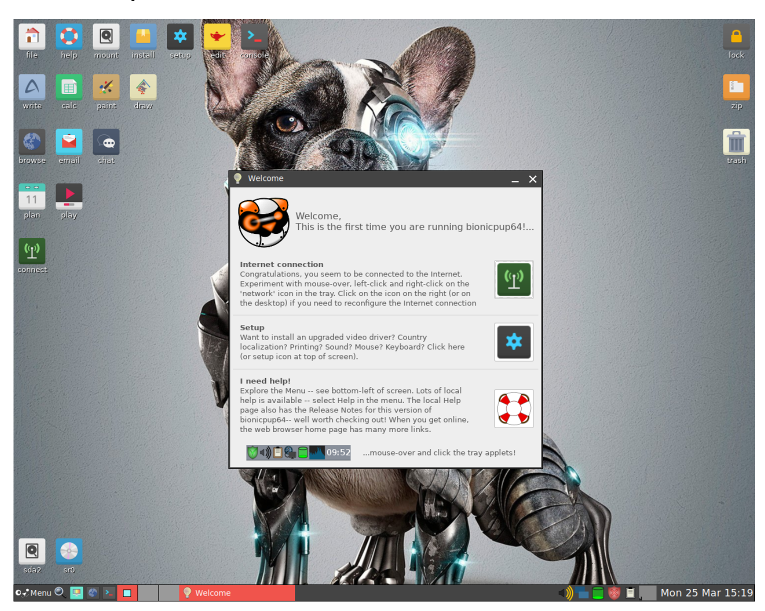
Al estar basada en Ubuntu, es totalmente

compatible con sus repositorios y paquetes. Desde la versión 0.4, incorpora una tienda de software propia, AppCenter, basada en el modelo “Paga lo que quieras” (Pay what you want, en inglés). Su interfaz se inspira en la funcionalidad y simplicidad de Mac OS X, aunque cabe destacar que está totalmente diseñada desde cero, y aspira a ser intuitiva para los nuevos usuarios sin consumir excesivos recursos.

10. Puppy – Reviviendo compus retros

Puppy Linux es una minidistribución portátil del sistema operativo GNU/Linux. Consiste en un CD autoejecutable con un Gestor de ventanas y programas suficientes para llevar a cabo la mayoría de tareas básicas en una computadora. Convierte a la máquina en una estación de trabajo con aplicaciones para conectar a Internet, navegar y chatear, procesador de texto, editores de imagen, audio y video, así como una variedad de

herramientas adicionales concentradas en un archivo ISO extraordinariamente pequeño y pese a ello está bastante bien equipada. Puppy se carga y ejecuta los programas en la memoria RAM de la computadora. Además, permite instalarse en discos duros o pendrives.

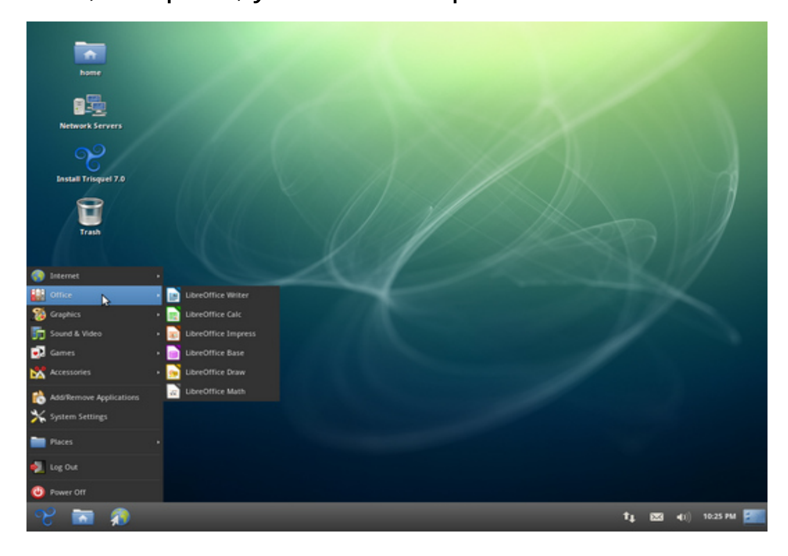


Puppy se carga en un disco RAM y a diferencia de las distribuciones de CD en vivo que tienen que seguir extrayendo sus cosas del CD, éste se carga en la RAM. Esto significa que todas las aplicaciones

comienzan en un abrir y cerrar de ojos y responden instantáneamente a lo que el usuario captura. Puppy Linux tiene la habilidad de arrancar desde una tarjeta flash o cualquier dispositivo de memoria USB, CD, DVD,etc.

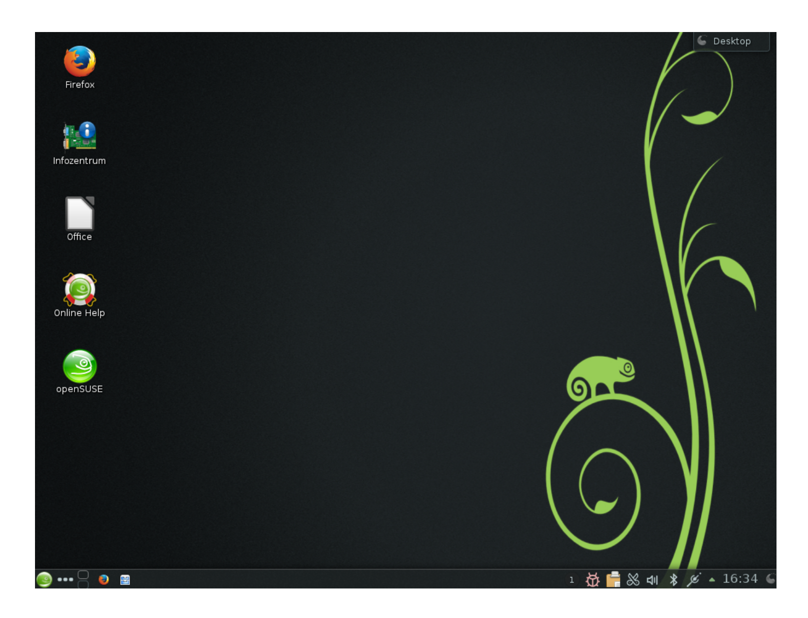
11. Trisquel – Libertad al 100%

Es una versión que utiliza el kernel Linux-libre. Los principales objetivos del proyecto son la producción de un sistema operativo totalmente libre, fácil de usar, completo, y con buen soporte de idiomas.



12. S.u.S.E – El camaleón informático

S.u.S.E. GmbH una empresa de informática alemana fue adquirida por NOVELL en el 2004 y para la sorpresa de TODOS decidió liberar el código, involucrando a toda la sociedad en el desarrollo del mismo. Fue un éxito TOTAL, sobre todo cuando nadie esperaba esto de una empresa tan privativa como NOVELL.



Una de las grandes desventajas de SuSE es que

solamente viene en ediciones X86-X64 y unas de sus ventajas es el YAST o sistema de instalación en un solo paso en donde la instalación es totalmente automática, detecta todo el hardware sin intervención alguna y sin reinicios en minutos hay un sistema OpenSuse funcional.

El 4 de noviembre de 2014 se publicó la versión 13.2, que entre sus novedades trae:

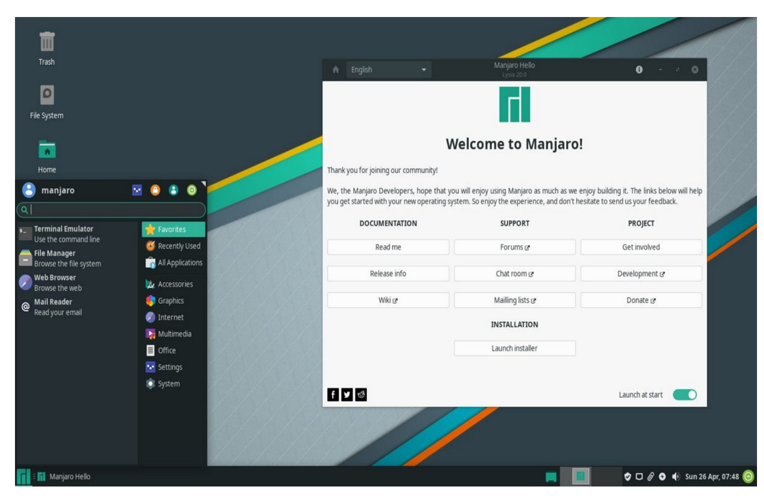
- Instalación en un sólo paso, no hace falta configurar nada en él tras el primer reinicio después de instalar.

13. Arch y Gentoo – Solo para expertos

Son dos distribuciones que se lanzaron en Marzo de 2002 y están orientadas a usuarios avanzados ya con cierta experiencia en sistemas operativos Linux. Ambas tratan de incluir solo programas necesarios y sin agregados innecesarios que permiten al usuario modelar el sistema de acuerdo a sus propias necesidades.



Vale destacar aquí a **Manjaro**, que si bien deriva de Arch, está enfocado en la facilidad de uso y es especial para novatos.



14. Distribuciones Argentinas...

14.1 Ututo: Es una distribución del sistema operativo GNU, que usa el núcleo Linux-libre. El nombre hace referencia a una especie de lagartija del norte de Argentina. Fue la primera distribución en ser reconocida como totalmente libre por el Proyecto GNU. En 2006, el proyecto UTUTO (que fue iniciado en la Universidad Nacional de Salta) fue declarado de Interés Nacional por la Honorable Cámara de Diputados de la Nación Argentina.

14.2 Tuquito: Es una distribución GNU/Linux proveniente de la ciudad de Tucumán. Implementa la tecnología LiveCD y realmente es muy sencilla para usuarios comunes.

14.3 Dragora GNU/Linux: Enfocada en la libertad, el idioma (castellano), la estabilidad y la seguridad; tiene la particularidad de haber sido desarrollada en Argentina sin basarse en ninguna otra distribución anterior. Es recomendada por la Free Software

Foundation, ya que es 100% software libre.

14.4 Lihuen GNU/Linux: Esta desarrollada por la Facultad de Informática de la Universidad Nacional de La Plata. Esta orientada a entornos de educación y oficina con aspecto similar a Windows.

14.5 Huayra GNU/Linux: Es una distribución pensada y desarrollada en Argentina para el uso de la comunidad educativa. Es el sistema operativo libre que desarrolla para los programas Conectar Igualdad y Primaria Digital para las netbooks que distribuye entre estudiantes y docentes de todo el país, desde el que se puede acceder a una gran variedad de programas y aplicaciones educativas.



*** Fuente consultada al principio de este capítulo:**

https://hipertextual.com/archivo/2011/08/20-curiosidades-sobre-linux/

LOS PROGRAMAS LIBRES MÁS REPRESENTATIVOS

Elizabeth Carolina Díaz Pepenal

LOS PROGRAMAS MÁS POPULARES DEL SOFTWARE LIBRE

En el mundo de la informática, muchas veces hablamos de programas y aplicaciones, pero alguna vez nos preguntamos que significa esto.

Una aplicación es un programa informático que permite a un usuario utilizar una computadora con un fin específico, además forma parte del software y suele ejecutarse sobre el sistema operativo.

A continuación se puede observar una clasificación de los programas más populares del software libre y en la área que se destacan.

Área	**Programa**
Ofimática	LibreOffice y OpenOffice.org
	AbiWord
Navegadores de Internet	Mozilla Firefox
	Chromium
	Apache

Multimedia	Audacity
	Banshee
	Rhythmbox
	Amarok
	Brasero
	VLC Media Player
	OpenShot
Diseñó Gráfico	GIMP
	Krita
	Tux Paint
	MyPaint
	LibreCAD
	Inkscape
	Blender
Comunicación	Pidgin
	Mozilla Thunderbird
	Empathy
Compresores de Archivos	7-Zip
	PeaZip

Descarga y Transferencia de Archivos	FileZilla
	eMule
	Ares Galaxy
	JDownloader
Los Básicos	Editor de Textos
	Calculadora
	Calendario
	Mapa de Caracteres
	Terminal

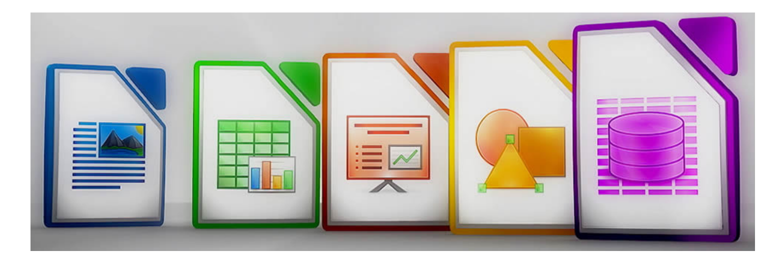
Programas de Ofimática

LibreOffice y OpenOffice

OpenOffice.org fue la suite ofimática de código abierto más popular por elección del usuario en un determinado periodo de tiempo, paso a llamarse Apache OpenOffice y se bifurco en LibreOffice.

Al hablar de LibreOffice, se puede decir que es una poderosa suite de oficina que incorpora varias aplicaciones potentes y que mediante su interfaz

gráfica permiten cultivar la creatividad del usuario logrando de esta manera una mejor productividad.



Entre las aplicaciones se destacan:

- **Writer:** tiene las características necesarias para trabajar con el procesamiento de textos y la autoedición.
- **Calc:** es la hoja de cálculo que resulta intuitiva y fácil de aprender.
- **Impress:** permite crear presentaciones multimedia, mediante diversos modos de edición y vistas, agregándoles efectos visuales, que permiten atraer la vista del espectador.
- **Draw:** Permite producir dibujos técnicos, carteles, etc. manipulando objetos 3D, gráficos, agruparlos, recortarlos y mucho

más.

- **Base:** posee una interfaz de base de datos y esta diseñada para satisfacer a una amplia gama de usuarios.
- **Math:** es el editor de fórmulas, que permite insertar fórmulas matemáticas y científicas.

A LibreOffice se lo puede descargar de la pagina: https://es.libreoffice.org/.

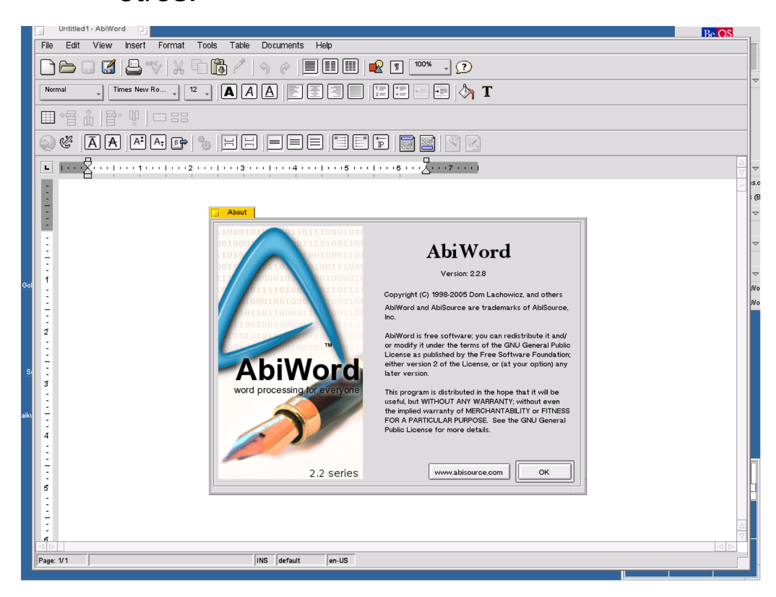
AbiWord

AbiWord es un completo procesador de texto, de software libre, multiplataforma, con licencia GPL, e incluso completamente gratuito.

AbiWord se caracteriza por:

- La ligera y sencilla interfaz gráfica.
- Permitir usarlo en equipos obsoletos.
- Poseer corrector ortográfico.
- Insertar imágenes, ecuaciones, cajas de texto, tablas, símbolos, hora, fecha, enlaces y otros.
- Dar formato básico al texto.

- Permitir exportar e importar documentos.
- Poder utilizarlo en sistemas operativos como Microsoft Windows, Mac OS X, GNU/Linux y otros.



Programas para Internet

Mozilla Firefox

Mozilla Firefox es un navegador de Internet libre y de código abierto descendiente de Mozilla

Application Suite.

Sus características más destacadas son:

- Existen versiones para Windows, Linux y Mac.
- Abre simultáneamente varias páginas web en pestañas independientes.
- Restaura la sesión cuando el navegador se cierra o reinicia accidentalmente.
- Ofrece un corrector ortográfico que subraya las palabras que no son tecleadas correctamente.
- No permite que una página web descargue, instale o ejecute programas en el equipo sin el consentimiento explícito del usuario.
- No almacena los datos privados de la navegación.
- Permite visualizar todo tipo de contenidos multimedia.

Chromium



Es un navegador Web de código abierto, a partir del cual se basa el código fuente de Google Chrome.

Este navegador es compatible con Windows, GNU/Linux y Mac OS.

Fue lanzado en el 2008 y cada día, los desarrolladores publican una nueva versión con pocos cambios.

Algunas características son:

- Ejecutar procesos por separado.
- Brindar seguridad y estabilidad al usuario.
- Adaptarse a los gustos del usuario.
- Mantener los ajustes y preferencias de otros navegadores de la PC.
- Permitir navegar de manera privada, sin mantener el historial de sitios visitados.
- Mayor rapidez en la navegación web.

Apache

Apache es una popular y eficiente alternativa, que ofrece servicios web.



El Servidor Web, es un programa especialmente diseñado para transferir datos de hipertexto, es decir, páginas web con todos sus elementos.

La ventaja de este programa, es que es un servidor web multiplataforma, es decir, puede trabajar con diferentes sistemas operativos y mantener su rendimiento.

Desde el año 1996, es el servidor web más popular del mundo, debido a su estabilidad y seguridad.-

Actualmente Apache sigue siendo desarrollado por la comunidad de usuarios que trabaja bajo la tutela de Apache Software Foundation.

Programas Multimedia

Audacity

Es una aplicación informática multiplataforma, libre, que se puede usar para grabación y edición de audio, distribuida bajo la licencia GPL.
Es el editor de audio más difundido en los sistemas GNU/Linux.



Se la utiliza mayormente, para crear y mezclar sonidos o grabaciones que están en los archivos de la computadora del usuario.
Las principales funciones de Audacity son:

- Grabación de audio a través de un micrófono o un mezclador.
- Digitalización de grabaciones de cintas.
- Importar archivos de sonido, editándolos y combinándolos con otros archivos o nuevas grabaciones.

- Permite agregarles nuevos efectos a la grabación.
- Visualiza las frecuencias y la amplitud de cada muestra, para poder analizar la reproducción completa.

Banshee

Banshee es un reproductor de audio que gestiona la biblioteca de música de la PC, emisoras de radio, integra tiendas de música y se sincroniza con dispositivos móviles y iPod.

Banshee es un reproductor multiplataforma disponible para los tres grandes sistemas operativos: Windows, OS X y Linux.

Su funcionalidad se basa en:

- La gestión de archivos multimedia.
- Permitir reproducir varias canciones o discos.
- Añadir radios manualmente.

- Descargar y escuchar gran cantidad de películas, libros, audiolibros, conciertos y conferencias.
- Dominio total en la reproducción de las canciones.

Rhythmbox

Rhythmbox es un reproductor de música que reproduce y ayuda a organizar la música digital. Fue originalmente inspirado por el reproductor de Apple, iTunes. Es software libre y actualmente está bajo desarrollo activo.

Se enfoca en una interfaz rápida y fácil de usar para buscar y reproducir música.

Se caracteriza porque permite:

- Buscar y reproducir música local de la biblioteca de la computadora.
- Escuchar radio en Internet.
- Crear listas de reproducción dinámicas.
- Reproducir y grabar CD de audio.

- Copiar música al iPod, iPhone o reproductor USB de almacenamiento masivo.
- Gestionar las colas de reproducción.
- Integración con otros programas programas externos, servicios y dispositivos.

Amarok

Amarok es un reproductor de audio libre desarrollado principalmente para el entorno de escritorio KDE. Usa el framework libre Qt por lo cual es multiplataforma.

Amarok no sólo es un reproductor de audio, ya que soporta de forma directa varios servicios de Internet y opciones especializadas para la gente que tienen grandes colecciones de música. aún así, resulta fácil de usar y su equipo trabaja mucho en la facilidad de uso de los menús y las opciones. Posee muchísimas características al igual que los anteriores reproductores ya vistos.

Brasero

Brasero es una aplicación de software libre de código abierto que se ejecuta exclusivamente en Linux, y que ofrece a los usuarios la capacidad de grabar música, datos y archivos de imagen de CD y DVD.

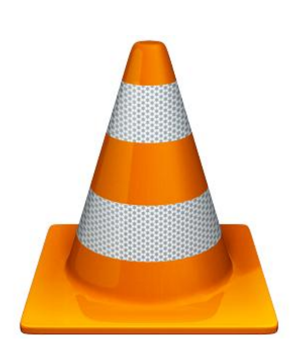


Se caracteriza por:

- Permitir editar el contenido del CD/DVD.
- Poder escribir una imagen virtual en la unidad principal.
- Permitir efectuar búsquedas de archivos de audio dentro de carpetas.
- Copiar CD/DVD.
- Copiar un CD/DVD al disco duro.
- Borrar CD/DVD.
- Poder guardar/cargar proyectos.
- Previsualizar imágenes, sonidos y vídeos.

VLC Media Player

VLC es un reproductor multimedia libre y de código abierto multiplataforma y que reproduce la mayoría de archivos multimedia.



Se caracteriza por:

- Permitir la reproducción y la descarga de vídeos en línea.
- Capturar un fragmento de un vídeo que se esté reproduciendo.
- Permitir grabar un vídeo del escritorio o grabar un vídeo con la cámara web.
- Guardar una copia de seguridad.
- Ofrecer canales de radios online.
- Añadir las propias marcas de agua cuando se reproduce un vídeo.

OpenShot

OpenShot Video Editor es un programa diseñado

para crear y editar vídeos en Linux. Fácilmente se puede combinar múltiples videoclips, sonido e imágenes en un solo proyecto y luego exportar el vídeo resultante a varios de los formatos de vídeo más comunes.

Este programa permite acceder a cualquier fotograma o parte del vídeo, y por lo tanto las secuencias de vídeo se pueden superponer, mezclar, y ordenar de manera muy creativa.



Las principales características que se pueden destacar son:

- Soporte de muchos tipos de vídeo, audio e imagen.
- Previsualización en tiempo real.
- Composición, superposición de imágenes y

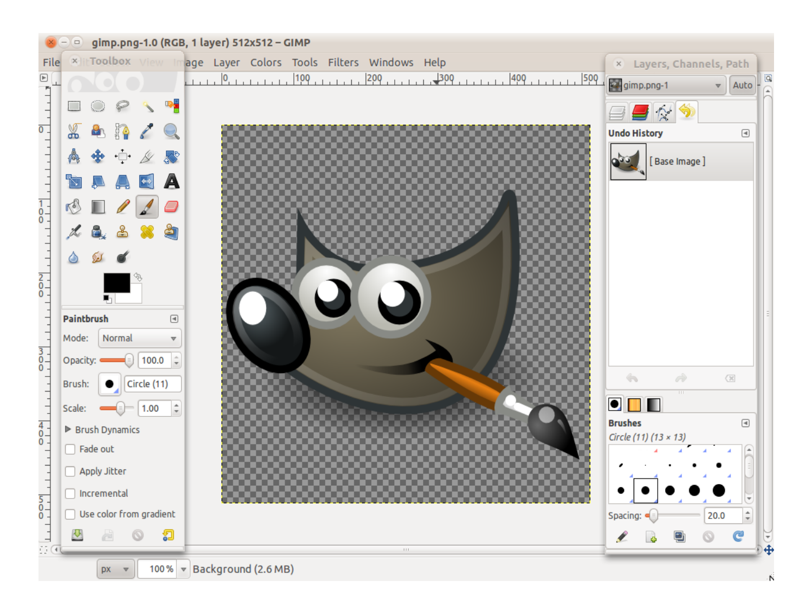
marcas de agua digital.

- Plantillas de título y creación de títulos.
- Créditos de película desplazables.
- Control manual sobre la velocidad y reproducción.
- Animación por secuencias de vídeo.
- Transiciones personalizadas.
- Mezcla y edición de sonido.
- Efectos digitales de vídeo.

Programas de Diseño Gráfico

GIMP

GIMP (GNU Image Manipulation Program) es un programa de edición de imágenes digitales en forma de mapa de bits, tanto para dibujos como para fotografías. Además es libre y gratuito. Forma parte del proyecto GNU y está disponible bajo la Licencia pública general de GNU y GNU Lesser General Public License.



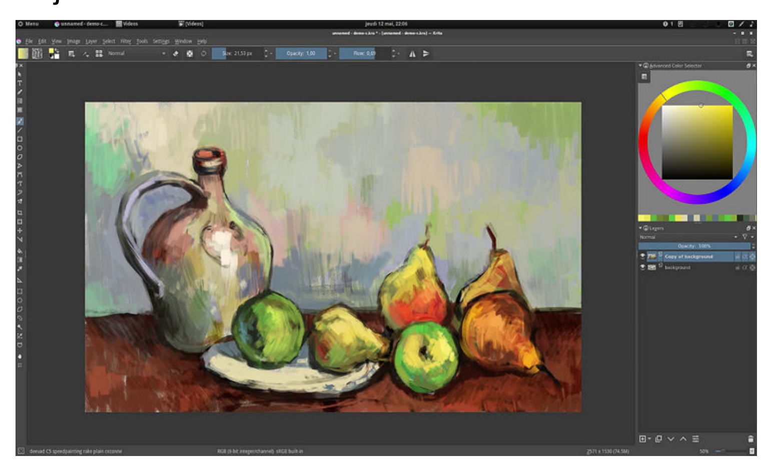
El programa funciona en los sistemas operativos: Unix, GNU/Linux, Microsoft Windows, Mac OS X, entre otros .

La interfaz se caracteriza por tener herramientas para el retoque y edición de imágenes.

Los desarrolladores y encargados se esfuerzan constantemente en mantener y desarrollar una aplicación gráfica de software libre, que sea de alta calidad para la edición y creación de imágenes originales.

Krita

Krita (cuyo significado es crayón en sueco) es el software de pintura digital e ilustración basado en las bibliotecas de la plataforma KDE e incluido en Calligra Suite. Diseñado como una suite de dibujo e ilustración digital, Krita es software libre distribuido bajo la licencia GNU GPL.



El diseño de Krita hace énfasis en la creación de imágenes desde cero más que en la manipulación de imágenes existentes. Soporta características no tan comunes en aplicaciones de dicho tipo, como por ejemplo la capacidad de trabajar tanto con

mapa de bits como ilustración vectorial, entre otras. A pesar de esto se centra en ofrecer características orientadas al dibujo digital más que a las funciones de edición fotográfica.

Tux Paint

Tux Paint es un programa de dibujo para niños de 3 a 12 años. Se lo usa como herramienta en el aprendizaje de dibujo por ordenador. Combina la interfaz con divertidos efectos de sonido y una mascota de dibujos animados que anima y guía a los niños que utilizan el programa.



La interfaz gráfica consta de un lienzo blanco y una variedad de herramientas de dibujo para ayudar a desarrollar la creatividad.

Se ejecuta en una variedad de plataformas, incluyendo todas las versiones de Windows, Mac OS X y Linux.

Este aplicativo se basa en software de código abierto, distribuido bajo la Licencia GNU General Public License y actualmente es desarrollado por voluntarios de todo el mundo.

MyPaint

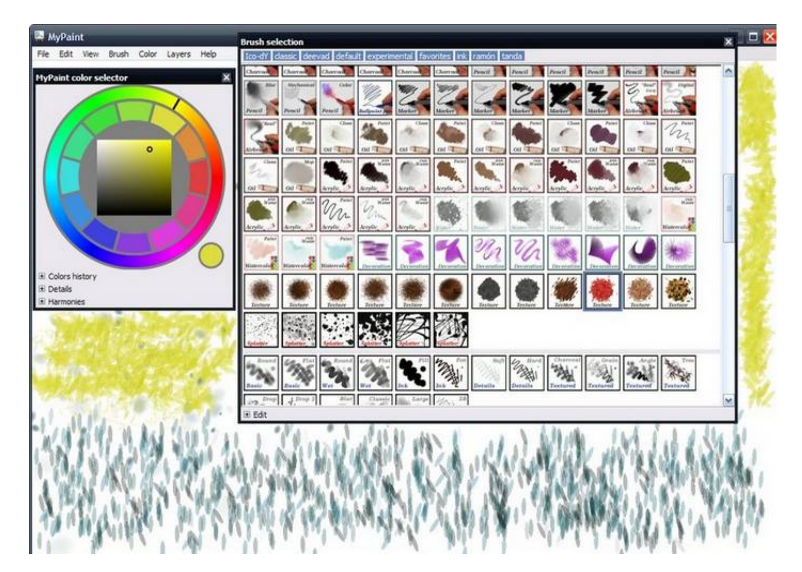
MyPaint es una herramienta para dibujar y pintar con una cantidad muy amplia de brochas.

La interfaz de MyPaint es minimalista, y completamente controlable, por ello normalmente se trabaja los dibujo sobre el lienzo .

Entre las características se destacan las siguientes:

- Funciona en Linux, Windows y Macintosh.
- Multitud de opciones para la creación y; configuración de la brocha.

- Lienzo ilimitado.



En conclusión MyPaint es una aplicación de software libre para ilustrar y dibujar de forma digitalizada.

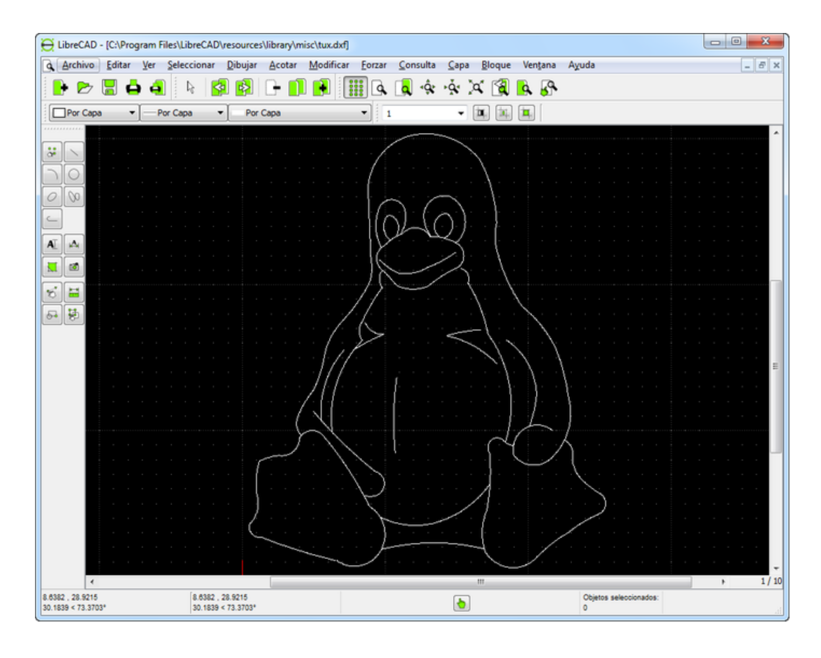
LibreCAD

LibreCAD, es una propuesta gratuita de código abierto, que brinda las herramientas básicas necesarias para empezar con el diseño gráfico.

Este es un proyecto que ofrece herramientas para diseñar y se distribuye bajo la licencia GPL.

Esta aplicación muchas veces es elegida por:

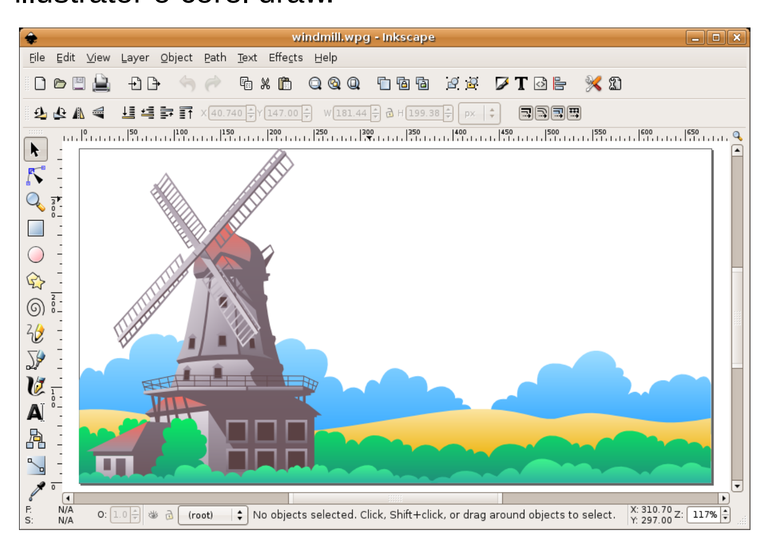
- ➢ La interfaz de usuario sencilla que posee.
- ➢ El panel de herramientas con el que cuenta es completo y esconde muchas más funciones de las que pueden apreciarse a simple vista.



LibreCAD aborda el dibujo 2D y se encuentra disponible para Microsoft Windows, Mac OS X y las principales distribuciones de GNU/Linux.

Inkscape

Inkscape es un editor de gráficos vectoriales de código abierto, con capacidades similares a illustrator o corel draw.



Esta aplicación trabaja con objetos y sus posibilidades son:

- Transparencias regionales y maestra para todos los objetos.
- Múltiples colores para seleccionar.
- Los objetos pueden agruparse o desagruparse.

- El objeto puede encerrarse en un borde de cualquier tamaño y color.
- Cualquier objeto puede ser duplicado indefinidamente.
- dibujar líneas de todo tipo.

En síntesis Inkscape es una herramienta de dibujo libre y multiplataforma para gráficos vectoriales.

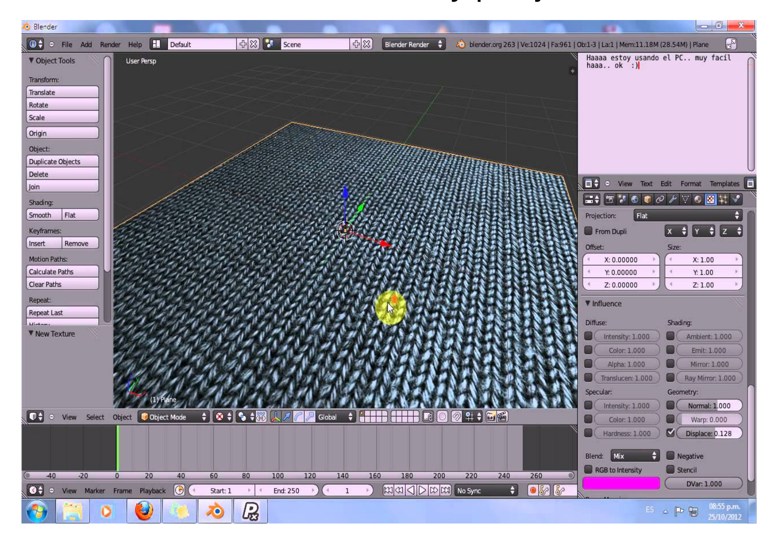
Blender

Blender es un programa informático multiplataforma, dedicado especialmente al modelado, animación y creación de gráficos tridimensionales.

Sus características más importantes son:

- Multiplataforma, libre y gratuito.
- Capacidad para una gran variedad de formas geométricas.
- Herramientas de animación.
- Edición de audio y sincronización de vídeo.
- Admite el lenguaje Python para automatizar o controlar varias tareas.

- ➢ Motor de juegos 3D integrado.
- ➢ Simulación de cabellos y pelajes.



Actualmente es compatible con todas las versiones de Windows, Mac OS X y GNU/Linux.

Programas para la Administración de la Comunicación

Pidgin

Pidgin es un sistema de mensajería instantánea multiplataforma, es decir, un programa capaz de

conectarse a diversas cuentas y redes de manera simultánea. Además de ser un software libre.

Este programa puede servir como alternativa a los diferentes servicios de mensajería instantánea presentes en la red, como Windows Live Messenger o Yahoo! Messenger. Su éxito radica en la posibilidad de tener en la misma agenda o lista de contactos a gente que use diversas cuentas de los proveedores de correo electrónico.



Entre sus características principales se destacan:

- La sencilla interfaz y que no incluye publicidades.
- Las conversaciones o la transferencia de archivos se presentan en pestañas.
- Registro de conversaciones.
- Reemplazar los nombres de los contactos de

la lista.

- Muestra un aviso o reproduce un sonido cuando un contacto se conecta/desconecta o cambia de estado.
- Soporte para WebCam.

Mozilla Thunderbird

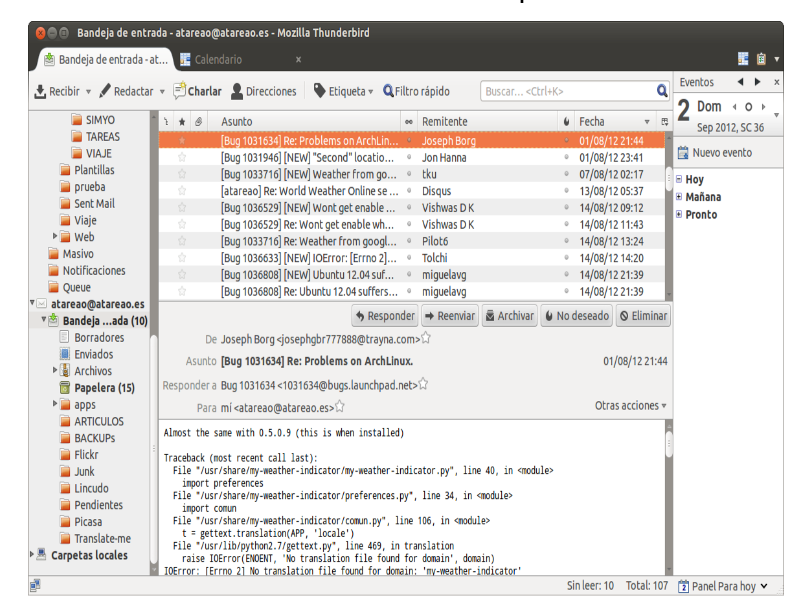


Thunderbird es una aplicación gratuita de correo electrónico que es fácil de configurar y personalizar.

Las características más sobresalientes de este programa son:

- Posee un asistente de configuración de cuentas de correo.
- Permite agregar direcciones de contactos de forma rápida y fácil.
- Advertencia a la hora de enviar un archivo adjunto.
- Cargar mensajes en pestañas separadas.

- Posee un administrador de actividad.
- Permite personalizar el aspecto de Thunderbird.
- Ofrece privacidad y protección al usuario.
- Protege de los fraudes por email.
- Filtra los correos basura o spam.



Empathy

Empathy es un cliente de mensajería instantánea basado en el entorno de comunicaciones Telepathy.

Entre las funciones principales se destacan:

- Multi-protocolo.
- Cambio a "ausente" de forma automática.
- Reconexión automática.
- Chats privados y públicos.
- Emoticonos y corrección ortográfica.
- Registro de conversaciones y búsqueda.
- Editar información de los contactos.
- Videoconferencia.

Programas Compresores de Archivos

7-Zip

7-Zip es una herramienta poderosa de compresión para Windows desarrollada por Igor Pavlov y se puede emplear en cualquier equipo.



Se caracteriza por:

- Comprimir y descomprimir archivos.
- Permitir administrar los archivos, mostrando fechas, tamaño, creación, modificación, último acceso, etc.
- Poner en prueba la capacidad de comprimir archivos del ordenador.
- Permitir crear archivos protegidos por contraseñas.

PeaZip

PeaZip es un programa open source muy completo para poder comprimir y descomprimir archivos, y en algunas funciones es más completo que los clásicos WinZip y Winrar.

Contiene además de las opciones comunes de cualquier compresor de ficheros, otras utilidades y opciones como por ejemplo: Eliminación segura de datos, poder cifrar y dividir archivos entre otras.



Programas para Descarga y Transferencia de Archivos

FileZilla

FileZilla es un programa multiplataforma diseñado por Tim Kosse. Es uno de los mejores software creados para gestionar archivos en el servidor web,

subir y bajar los archivos ordenándolos en capetas, también permite renombralos y realizar muchas acciones más.



Al ser un software libre, bajo la Licencia Pública General de GNU, es utilizado para la transferencia de archivos entre PC y un servidor web, ya que se basa en la arquitectura cliente–servidor.

eMule

En la madrugada del 13 de mayo de 2002 nació el Proyecto eMule, cuyo principal propósito fue el intercambio de archivos del tipo "peer-to-peer" (de amigo a amigo) en todo el mundo, además es gratuito.



Algunas funciones son:

- Garantiza que se encuentre el archivo buscado.
- Comprobación de los archivos a lo largo del proceso de descarga.
- Asegura la integridad de los datos.
- Permite crear categorías para organizar las descargas.
- Ofrece una amplia gama de posibilidades de búsqueda.
- Posee sistema de Mensajería.
- Incorpora un chat para contactar con usuarios de todo el mundo.

Ares Galaxy

Ares es un programa que permite buscar y descargar cualquier archivo ya sea música mp3, vídeos, programas, documentos, etc.



Posee un sistema de descarga de archivos que busca automáticamente en diversas fuentes el archivo solicitado, permitiendo descargar, el mismo archivo simultáneamente desde diferentes partes de todo el mundo.

El programa posee un reproductor que permite previsualizar archivos de audio y vídeo mientras son descargados, además de crear listas de reproducción con vídeos y temas musicales.

El Ares también cuenta con salas de chat y permite que el usuario puede crear su propia sala de chat.

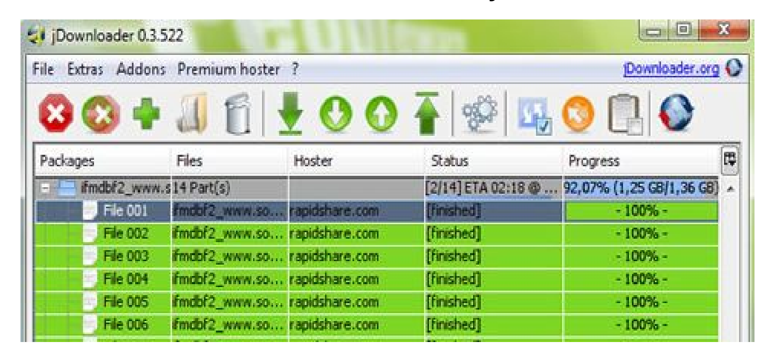
A pesar de ser un software libre solo esta disponible para sistemas Windows.

JDownloader

JDownloader es una herramienta gratuita, de código abierto de gestión de descargas, fácil y rápida.



Los usuarios pueden iniciar, detener o pausar descargas, estableciendo limitaciones de ancho de banda y mucho más.



Se destaca por:

- Permitir descargas múltiples.
- Ser compatible con múltiples portales.
- Funcionar como gestor de descargas convencional.

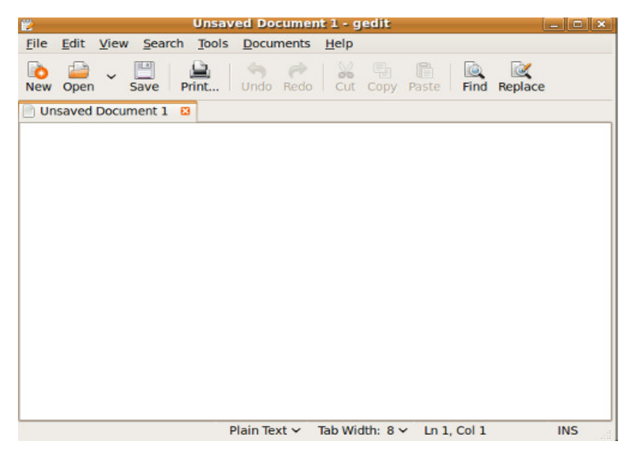
- Continuar con descargas pausadas.
- Una interfaz amigable.

En síntesis es una plataforma de código abierto diseñada para simplificar la descarga de archivos de los servidores.

Programas Básicos de Linux

Editor de Textos (Gedit)

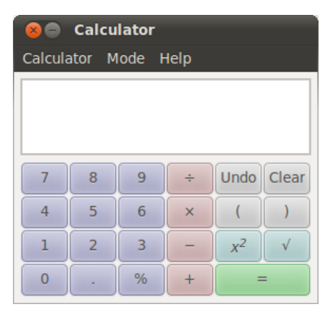
Es un programa informático que permite crear y modificar archivos digitales compuestos únicamente por textos sin formato, conocidos comúnmente como archivos de texto o “texto plano”.



El programa lee el archivo e interpreta los bytes leídos según el código de caracteres que usa el editor. Seria como un bloc de notas para tomar apuntes.

Calculadora

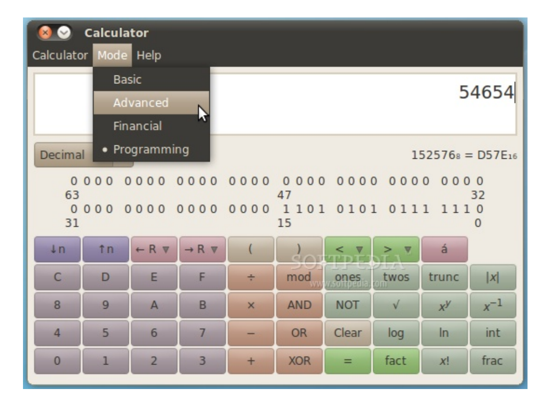
Es una herramienta empleada para los cálculos



numéricos y no es necesario disponer de un ordenador potente, ni de un software demasiado avanzado.

Existe en el modo básico y científica.

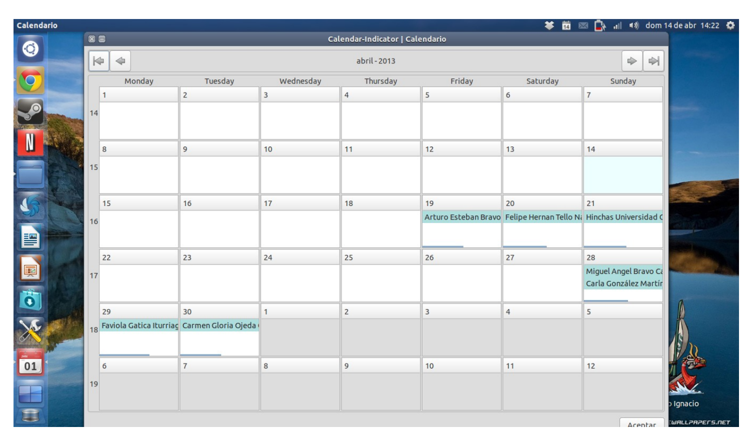
La científica que contiene logaritmos, conversiones de la base numérica, operaciones lógicas, radianes, grados y centesimales.



Calendario

Es un calendario de escritorio en el que se puede anotar todas las fechas y citas importantes, así como disfrutar de una imagen distinta como fondo de escritorio cada mes del año.

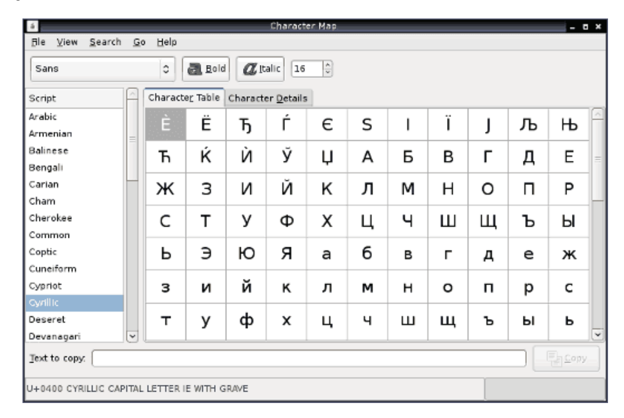
El programa muestra un calendario mensual que puedes colocar en el rincón de la pantalla y que permite su personalización cambiándole el tamaño, color, formato de fecha, etc.



Mapa de Caracteres

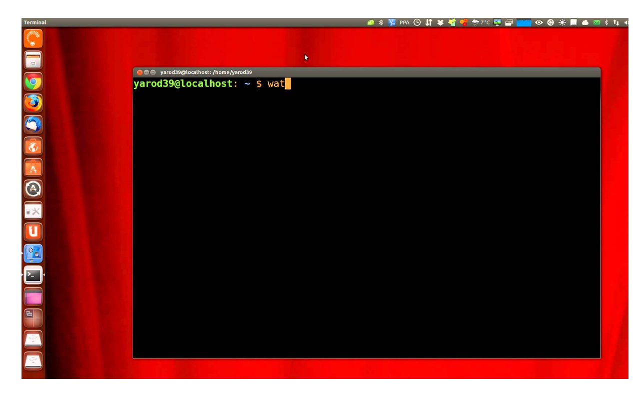
Mapas de caracteres se puede usar para encontrar cualquier símbolo y carácter especial desde

cualquier fuente instalada en el sistema Linux.



Terminal

La consola o terminal es un programa informático donde interactúa el usuario con el sistema operativo mediante una ventana que espera ordenes escritas por el usuario desde el teclado.



Otros Destacables

Existen varios programas libres que se pueden destacar como por ejemplo los editores de texto **Emacs** y **Geany**, los reproductores de audio **Audacious** y **Clementine**, el reproductor de vídeo **Totem**, la aplicación para DJ **Mixxx**, los lectores de PDF **Evince**, **Okular** y **Sumatra**, el grabador de discos **K3b**, el cliente BitTorrent **Transmission**, el navegador **Tor**, el gestor y organizador de libros electrónicos **Calibre** y otros utilitarios como **Kodi**, **Kdenlive**, **Scribus**, **Wordpress**, **Cheese**, **GParted**, **Etcher**, **Yumi**, **Rufus** y **Unetbootin**.



LA LIBERTAD EN LA LENGUA EXTRANJERA INGLÉS

EMILCE ISABEL RODÓ

La Libertad en el Inglés

Este capítulo se advoca a la vertiginosa tarea de enseñar y aprender inglés desde la libertad que nos garantizan los programas de software libre como Anki, Anagramarama, GoldenDict y Moodle. Los mismos fueron seleccionados de la fuente de proyectos de software más grande https://sourceforge.net y de otras fuentes confiables.



Antes de mencionar las características de los softwares cabe resaltar la evolución de la lengua extranjera inglés, en cuanto a metodología, porque los programas que se presentarán complementan al estudio formal del idioma inglés desde el enfoque comunicativo.

Desde el Grammar Translation Method, un método tedioso que proponía como eje en las clases traducciones, se remitía a lo gramatical y en su mayoría se remitían la memorización de frases estructuradas, la enseñanza de la lengua extranjera inglés fue atravesando distintos métodos que surgían en reacción al anterior. Se distingue así, direct method, audiolingual method, the structural approach, suggestipedia, total physical response, hasta llegar al método usado a nivel mundial que es el Communicative Approach (método comunicativo), aprendemos inglés para comunicarnos.

Entonces podemos preguntarnos: ¿Qué libertad le

da el software libre a la lengua extranjera? El idioma inglés se ha liberado porque ha ido atravesando por una metamorfosis de métodos asegurándonos la libertad de expresarnos, de conectarnos, de establecer relaciones internacionales, de crecer intelectualmente y por sobre todo abrir las mentes a nuevas perspectivas sociales. Entonces, ¿Porqué vamos a ir en contra de la filosofía de la libertad? Siguiendo éstas corrientes de emancipación podemos establecer un vínculo intrínsecamente noble al momento de relacionar el idioma inglés con software libre. Como el aprendizaje de toda lengua extranjera propone libertad de pensamiento, libertad social, y por sobre todo manifiesta la libertad del ser, usando software libre complementamos y aseguramos la filosofía de independencia social.

Dado que estamos insertos en la era tecnológica y muchos de los estudiantes son nativos digitales y que para la mayoría de nosotros los métodos

tradicionales (como la lectura de libros o la repetición de palabras) ya no son factibles para estudiar un idioma, con este aporte recomendamos programas de software libre para promover el aprendizaje del idioma inglés de una forma más innovadora, práctica y dinámica, pero por sobre todo *libre*. De esta manera podemos adentrarnos a los beneficios que podemos obtener del software libre porque si de libertad[2] hablamos podemos decir que cada programa que se presenta puede ser modificado a las necesidades del aprendiz y distribuidos a toda la comunidad.

Los programas que se presentan a continuación poseen características específicas para fomentar el desarrollo de una/s de las macrohabilidades

2Cuatro libertades del software libre. Libertad 0: Poder usar el programa con cualquier propósito.
Libertad 1: Poder estudiar cómo funciona el programa y poder modificarlo.
Libertad 2: Poder distribuir copias del programa.
Libertad 3: Poder mejorar el programa y poder compartir dichas mejoras para beneficio de todos.

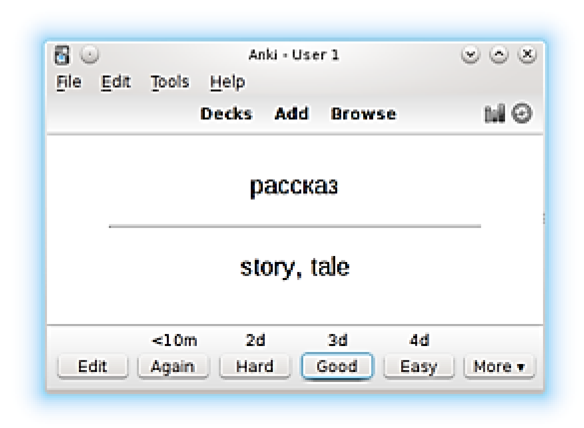
lingüísticas de la lengua extranjera inglés, como la escucha, el habla, la lectura y la escritura.

Anki

Anki es un programa que permite el aprendizaje de inglés mediante tarjetas (flashcards), en las que se puede insertar imágenes y descripción escrita del concepto de cada palabra a aprender. Una vez que se han insertado todos los requerimientos se inicia la práctica y al principio sólo se muestra la imagen o palabra, mentalmente el estudiante recordará el concepto y luego haciendo click en ‘good’, si pudimos recordar la palabra en un tiempo prudente, ‘easy’, si se recordó la palabra de forma fácil o ‘hard’ si tomo tiempo recordar la palabra, se mostrará la información de la tarjeta para chequear la factibilidad del aprendizaje.

Al tiempo de intervalo de presentación de tarjetas se lo agrega al inicio y con el transcurso de la presentación se da la opción de volver a ver la

tarjeta de acuerdo al tiempo que se le asigne.



Características de Anki:

- Utilizando su página web 'Ankiweb' se puede mantener el software actualizado y tener el contenido en varios equipos. Entonces, se puede *sincronizar*.
- Se puede personalizar porque es *flexible*.
- Permite el uso de *multimedia*. Se puede agregar videos, audios, imágenes y también permite marcar las tarjetas con control de muestra.

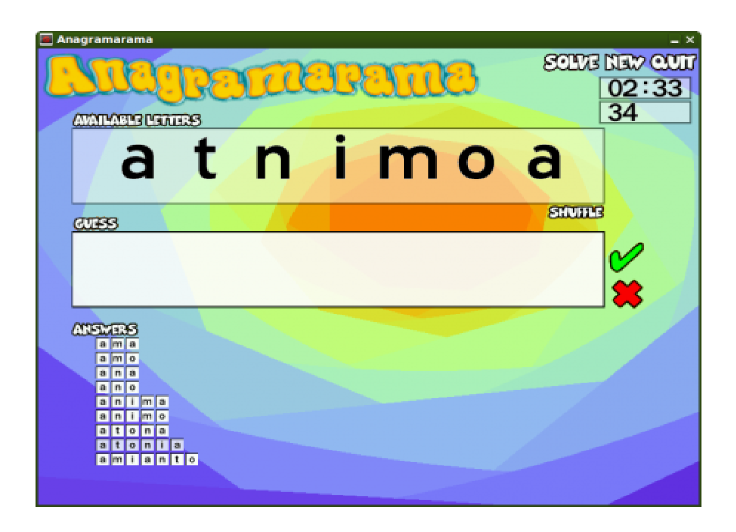
- Anki permite crear hasta 100mil tarjeta de manera *óptima*.
- Se pueden agregar nuevas características haciéndolo una herramienta completa e innovadora a los usuarios porque es *código abierto.*
- Con Anki se pueden aprender otras disciplinas aparte de inglés como:
  - Cualquier idioma
  - otras áreas como medicina y derecho
  - nombres y rostros
  - destinos geográficos: países, ciudades, pueblos
  - acordes de guitarra

Se puede descargar Anki en https://apps.ankiweb.net/

ANAGRAMARAMA

Este programa es un juego que provee 7 letras de las cuales hay que formar palabras de la extensión

de tantas letras como casilleros hayan en cada línea. La interfaz tiene el cronómetro con 5 minutos al término de los cuales se muestran todas las palabras formadas. Al momento de ir formando palabras el programa marca con una tilde verde si se formó una palabra correcta (ubicándola en un casillero) y con una cruz roja si las letras no forman ninguna palabra. Anagramarama se encuentra configurado en inglés pero puede ser modificado al español de manera manual. Este software está en distribuciones GNU/Linux.



Anagramarama puede ser un complemento luego de haber aprendido palabras en Anki. Podemos hacer una autoevaluación del contenido léxico que hayamos adquirido anteriormente.

El nivel del juego es tanto para principiantes como para profesionales porque lo que se evalúa es el léxico de la persona y el nivel de inglés (o de otro idioma) dependerá de las palabras formadas.

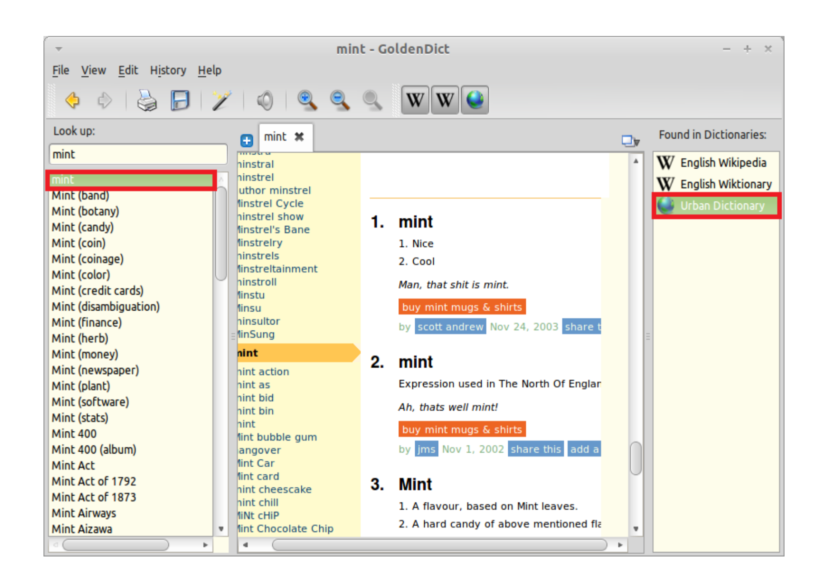
Ventajas de usar Anagramarama:

- Adiestrar la mente al escribir diferentes palabras de distinta longitud dadas las 7 letras.
- Afianzar el léxico adquirido y desarrollar la creatividad lingüística.
- Promover el uso de tics en juegos.
- Desarrollo de la coordinación óculo-manual.

Se puede descargar Anagramarama en https://download.cnet.com/Anagramarama-for-Linux/3000-2111_4-75210104.html

GoldenDict

Este software libre es una compilación de diccionarios y páginas web (www.wiktionary.org y www.wikipedia.org) en un solo diccionario electrónico. Está disponible en Linux y FreeBSD. Tiene una interfaz flexible que se puede configurar. GoldenDict se encuentra en las lenguas de quechua, aimara y castellano.



Este diccionario electrónico permite hacer la

búsqueda de una palabra en todos los diccionarios y páginas web que se le inserten al programa. Entonces, se tendrán los conceptos y acepciones de todos los diccionarios compaginados en GoldenDict.

Se destaca por las siguientes ventajas:

- La velocidad de sus búsquedas.
- Se inserta o desactiva un diccionario de manera muy rápida por su interfaz flexible.
- Descarta caracteres como signos de puntuación centrando la búsqueda en la palabra.
- Variedad en la configuración de idiomas en la que se presenta GoldenDict.
- Se pueden agregar páginas web como wikipedia.org y wiktionary.org da una amplitud en el concepto y ejemplos que pueda tener la palabra en diferentes usos.

Cabe destacar que este programa puede ser usado

offline lo que es un beneficio, pero también no se obtendrá la misma amplitud del concepto.

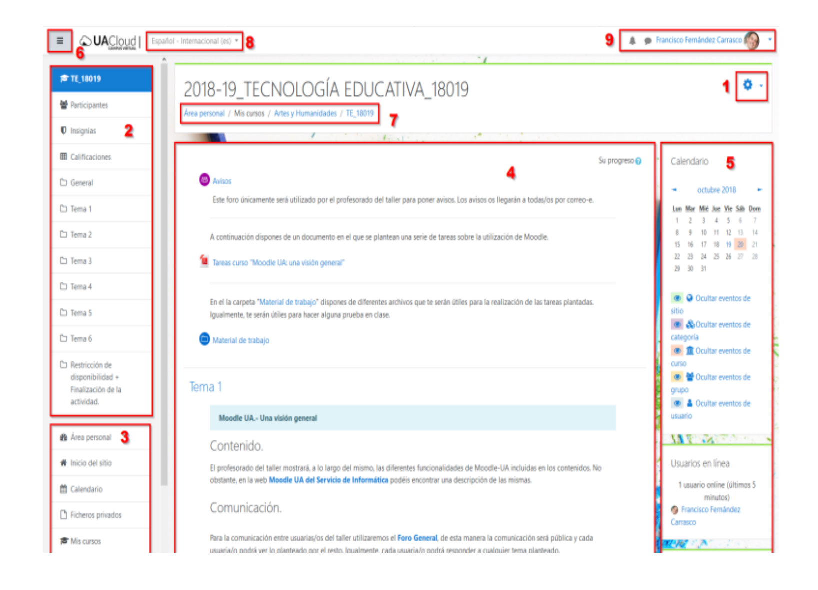
Se puede descargar GoldenDict en https://sourceforge.net/projects/goldendict/

Moodle

Moodle es una de las plataformas más usadas en educación a distancia, como complemento de educación presencial formal y como soporte de universidades para llevar el registro de sus estudiantes. Este software posibilita que educadores y estudiantes se puedan encontrar para intercambiar diálogos, conocimiento, puestas, etc. en una instancia virtual. Este software permite crear cursos, clases y/o encuentros en línea y fuera de línea también dado que la interfaz es flexible, configurable y completa.

El acrónimo Moodle se refiere originalmente es de Modular Object-Oriented Dynamic Learning

Environment que en español significa Entorno de Aprendizaje Dinámico Orientado a Objetos y Modular, este programa promueve el aprendizaje sobre la pedagogía social constructivista por lo que es la plataforma más elegida por educadores por sobre otros softwares.



Cabe destacar las ventajas de Moodle

El principal beneficio, tanto de Moodle como de Anki, Anagramarama y GoldenDict es que son

Código Abierto[3]. Estos softwares tienen Licencia Púbica GNU, lo que quiere decir que pueden ser usados sin pagar licencia, se los puede modificar, usar copiar y distribuir promoviendo el trabajo colaborativo por un mejor producto a beneficio de todos. Por esta razón Moodle está en constante evolución siendo flexible y de fácil configuración.

Se puede descargar Moodle en https://sourceforge.net/projects/moodle/

Conclusión

En este capítulo fue inspirado en la libertad que como sociedad nos merecemos y mi aporte desde la lengua extranjera fue para llegar a todas las personas que quieran romper paradigmas y estructuras sociales porque somos seres libres. Promoviendo el software libre aportamos a la

3 El **código abierto** es un modelo de desarrollo de software basado en la colaboración **abierta**. Se enfoca más en los beneficios prácticos (acceso al **código** fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre https://es.wikipedia.org/wiki/C%C3%B3digo_abierto

hermandad y al valor humano como personas porque la construcción del software la hacemos entre todos desde la profesión y/o lugar que ocupemos en la sociedad.

Luego de haber tenido una breve reseña sobre la lengua extranjera inglés, podemos reflexionar como el conocimiento nos hace libre y practicarlo nos inspira a un nuevo futuro . Por eso desde este apartado se puede pensar que así como la lengua extranjera y el software libre pueden ir de la mano también lo puede hacer desde otras áreas por eso desde donde estén se los invita a brindar su aporte y fomentar el trabajo colaborativo para promover libertad.

Fuentes consultadas:

https://www.articaonline.com/2016/09/moodle-software-libre-para-la-gestion-del-aprendizaje/
https://download.cnet.com/Anagramarama-for-Linux/3000-2111_4-75210104.html
https://www.wikizero.com/en/Anki_(software)
https://www.ihaveapc.com/2011/10/how-to-install-and-use-goldendict-dictionary-software-in-linux-mint-ubuntu/
https://sourceforge.net/projects/goldendict/
https://sourceforge.net/projects/moodle/

HACIA LA INDEPENDENCIA TECNOLÓGICA

Natalia del Valle Rodríguez

Huayra: el sistema operativo creado por argentinos y para el Mundo

Introducción

El Sistema operativo Huayra fue creado en nuestro país por CENITAL (Centro Nacional de Investigación y Desarrollo de tecnologías libres) dirigido por Javier Castrillo[4], en el marco del programa conectar igualdad.

Nace con el objetivo de brindar herramientas nuevas para mejorar la calidad educativa de nuestros alumnos y lograr la igualdad de oportunidades en el marco del conocimiento tecnológico. Por sus características de software libre, Huayra no solo inicia el camino hacia el desarrollo de sistemas operativos propios, en

4CENITAL es un área del "***Programa Conectar Igualdad***" que nace con la idea de generar y motorizar experiencias de investigación y desarrollo en el campo de las Tecnologías Libres en nuestro país, para contribuir con la soberanía e independencia tecnológicas y apoyar la gestión de la Administración Pública y de la Comunidad. Extraído de Dirección General de Educación Superior de Córdoba. Para más información ir a: http://dges.cba.infd.edu.ar/

nuestra región, sino que también se posiciona bajo el enfoque de la soberanía tecnológica.



Soplan "Vientos de Cambio"

En este artículo intentaremos exponer algunas de las características más sobresalientes del proyecto Huayra, como así también algunas experiencias concretas que se llevaron adelante en distintos Establecimientos educativos.

Conectando oportunidades

No podemos describir al proyecto Huayra sin antes referirnos al Programa Conectar Igualdad. En breves líneas diremos que surge en nuestro país

como un proyecto educativo destinado brindar una respuesta efectiva a los desafíos que se presentan en el mundo actual, en materia de Inclusión Digital y de calidad educativa, en el contexto de las nuevas Tecnologías de la Información y la Comunicación. Se puso en marcha a partir del año 2010 a través del decreto 459 del poder Ejecutivo, por el cual quedaba establecida la decisión, del gobierno nacional, de distribuir computadoras portátiles a todos los alumnos y docentes de las escuelas públicas, tanto del nivel Medio como Niveles de Educación Superior y Escuelas de Educación Especial. El nivel primario, por su parte, obtendría los beneficios de Conectar Igualdad recién a partir del año 2014 con la entrega de las Aulas Digitales Móviles.



Los objetivos generales del programa apuntaban hacia la posibilidad de reducir la brecha digital educativa en el país, mejorar la calidad de la educación en la Escuela Pública y lograr la inclusión digital en términos de federalismo. Se presumía que el programa no sólo debía corresponderse con la entrega y manejo de equipamiento tecnológico e infraestructura, sino que además tenía que estar acompañado de propuestas pedagógicas y acciones de formación concretas, destinadas a promover y sostener el uso de las TICS en la educación Secundaria y al mismo tiempo brindar apoyo, a los docentes, en el uso de las computadoras portátiles en el aula. Por ello resultaba necesario generar dispositivos de capacitación, cuyos destinatarios directos fueran todos los profesionales y personal de apoyo en actividad, tanto de escuelas públicas como privadas de las 24 jurisdicciones, como así también estudiantes de formación docente. Un ejemplo de

esta iniciativa fue la creación del Programa nacional Nuestra Escuela que con el respaldo del Consejo Federal de Educación promueve actualmente la formación gratuita y universal para todos los docentes del país[5].

En palabras de los responsables directos del Programa podemos afirmar que: “Conectar Igualdad surgió como una propuesta de trabajo destinado a "lograr una sociedad alfabetizada en las nuevas Tecnologías de la Información y la Comunicación, y que permite el “acceso democrático a recursos tecnológicos e información sin distinción de grupo social, económico ni de las más diversas geografías, tanto urbanas como rurales”[6].

Finalmente, al constituirse como un proyecto

5El programa cuenta con un abanico de ofertas educativas relacionadas tanto con los diversos campos disciplinares como con propuestas educativas para el uso de TICS en el aula. Cabe recordar que surge sobre las estructuras de las denominadas escuelas de innovación educativa.

6¿Qué es Conectar Igualdad? www.conectarigualdad.gob.ar 9 de febrero de 2016 http://www.conectarigualdad.gob.ar/seccion/sobre-programa/que-conectar-igualdad-53

educativo dependiente de las políticas públicas del Estado nacional, Conectar Igualdad se financió hasta el año 2015 con recursos de la Administración Nacional de Seguridad Social (ANSES), en un trabajo articulado con Ministerio de Educación de la Nación y el Ministerio de Planificación Federal de Inversión Pública y de Servicios. En términos cuantitativos ya llevan entregadas más de 5 millones de netbooks en alrededor de 9.898 escuelas de todas las provincias argentinas, sin mencionar el equipamiento tecnológico que acompaña la distribución de estas computadoras por alumnos.[7]

Huayra Linux: el sistema operativo libre de Conectar Igualdad.

La novedad más significativa, si me permiten la afirmación, que podemos observar en el programa

7Datos extraídos del portal www.conectarigualdad.gob.ar última actualización 9 de febrero de 2016 http://www.conectarigualdad.gob.ar/noticia/la-presidenta-entrego-la-netbook-millones-2367

conectar igualdad es que las computadoras portátiles cuentan con un doble sistema operativo: Microsoft Windows y GNU Linux. El alumno o cualquier usuario puede acceder a cualquiera de estos dos sistemas operativos, de una manera sencilla y según sea su preferencia.



Soberanía Tecnológica y Calidad Educativa

La necesidad y decisión estratégica de instalar dos sistemas operativos resulta muy significativo por dos razones:

1- porque nos permite explorar ambos sistemas y

de allí la posibilidad de elegir libremente cuál de los dos queremos usar y,

2- porque nos permite optar entre el uso de un software propietario (Windows) o un software libre (Linux).

Reconocemos fehacientemente que el número de usuarios que hoy en día utilizan Windows es mayoritario y está por fuera de toda discusión más allá de los análisis que surjan de ello. Los mismos alumnos y docentes beneficiarios de las Netbooks prefieren el uso de Windows, antes que cualquier otro sistema operativo. Sin embargo, cada vez son más las personas que progresivamente comienzan a explorar otros sistemas operativos como es el caso de Linux. De hecho, la decisión de trabajar con sistemas doble booteo estimula la curiosidad y permite explorar otros terrenos casi desconocidos y abrir nuestros conocimientos a otros tipos de software, quizás, no tan explotados comercialmente.

A partir del año 2012 la iniciativa argentina, redobló sus esfuerzos y se orientó hacia la creación y distribución de un sistema operativo propio: Huayra Linux. Este sistema operativo con código abierto fue creado desde y para el programa conectar Igualdad por programadores argentinos. Al ser un software libre, huayra, no depende de las corporaciones trasnacionales, reduce los costos de fabricación y las mejoras dependen no de un solo fabricante, sino que permite el trabajo en conjunto con sus usuarios permitiendo el crecimiento del mismo y evitando así todo tipo de monopolios. En definitiva, se enmarca dentro de los principios de soberanía tecnológica y promoción de la industria científica y nacional en la que se encuadra nuestro país a partir de los últimos años.

Las ventajas educativas que posee su inicio por defecto en las netbooks, es que podemos crear, innovar, modificar, y abrir caminos para nuevas producciones en materia de tecnología y

Comunicación. Su utilización nos permite no tan sólo estudiar su funcionamiento sino también modificarlo de acuerdo a nuestras necesidades y capacidades. Huayra sigue la lógica de las investigaciones y publicaciones académicas donde los conocimientos se construyen y divulgan sirviendo de base para nuevas producciones. Contar con este tipo de sistema operativo propio permite a la Industria Nacional avanzar sin escalas en el ritmo del desarrollo tecnológico y fomentar la soberanía tecnológica en materia de innovación educativa. A la par de colocarnos en un lugar privilegiado como productores de tecnología y no en simples consumidores.

En el ámbito escolar, Huayra se presenta como una herramienta pedagógica que permite la construcción colectiva de los conocimientos. Nos encontramos con programas gratuitos y libres para la edición de textos, audios, videos, y programas para educación especial. Al ser un software libre

sus aplicaciones permiten desplegar la creatividad en nuestros alumnos y modificar todo aquello que crean conveniente para mejorar estas aplicaciones o, en su defecto, crear nuevos dispositivos de aprendizaje. En este sentido son ampliamente ventajosas las oportunidades operativas del programa porque permiten ajustarse a las necesidades tecnológicas según la región o el lugar donde uno se encuentre. Por ej. un sistema operativo propietario no nos permitiría modificar el idioma en un procesador de texto, en cambio Huayra al ser un software libre permite difundir su código de acceso y operadores en red pueden realizar todo tipos de ajustes e incorporar nuevos idiomas necesarios por ej. para las comunidades originarias de nuestro país.

En definitiva el sistema Huayra junto con el programa conectar igualdad se encaminan hacia la democratización del conocimiento mediante la igualdad de oportunidades y la inclusión educativa.

Experiencias educativas mediante el uso de Huayra

De acuerdo a declaraciones y/o apreciaciones de portales dedicados a la comunidad tecnológica, Huayra se presenta como una herramienta educativa que no limita espacio ni calidad en la selección de software en su sistema operativo. La sencillez en el uso de estos recursos junto con los tutoriales hace de Huayra un instrumento ventajoso tanto para usuarios especializados como para los inexpertos en el área.

A través de Huayra nos acercamos a los contenidos de Wikipedia a través del programa ***CdPedia***, observamos el sistema solar a través de ***Celestia*** y ***Stellarium***, aprendemos de cálculos matemáticos y geografía con ***Geogebra*** y ***Dr Geo***, trabajamos con moléculas, estructuras y elementos químicos con ***Avogadro*** y ***Jmol***, aprendemos dactilografía con ***Klavaro*** y ***KTouch***, desarrollamos actividades educativas multimedias con ***JClic***, aprendemos

jugando con los programas de la suite educativa ***GCompris*** y desde *Mi Escritorio* podemos vincularnos con contenidos del portal **Educ.ar** sin necesidad de estar conectados a internet.



En cuanto a experiencias concretas en comunidades educativas compartimos las actividades de dos escuelas: una en la provincia de Salta y otra en Río Negro:

Democratización en la esfera de la información y la Comunicación: "Programas de radio" En primer lugar mencionaremos la experiencia desarrollada por la Profesora Fernanda Planels de la Escuela Agrotécnica Nº 3151 de Payogasta, Salta. La

actividad que llevó adelante esta docente, junto con sus alumnos, fue el desarrollo de programas de radio mediante el manejo de **Audacity**. El proyecto significó una tarea integral, donde los alumnos debían diseñar el itinerario para el programa de radio, seleccionar la música adecuada y establecer criterios para los tiempos de duración de cada una de las partes del programa radial. Lo significativo de la actividad fue lograr que alumnos provenientes de los parajes del interior de nuestra provincia se animaran a participar y "hablar en público", como así también buscar y seleccionar información posible de ser difundidos en un programa radial. Esta actividad se hizo extensiva a otras instituciones educativas y se puede observar cómo y de qué manera las netbooks del programa conectar igualdad ponen a disposición de la comunidad educativa programas que permiten la democratización de las comunicaciones mediante el manejo sencillo de software libre.

Otras experiencias nos enseñan no solamente a explorar o utilizar las aplicaciones libres sino también la posibilidad de crear nuevos dispositivos.

En la escuela Técnica Nº 18 de Villa Regina, provincia de Río Negro, el profesor Cristián Liberatore junto con sus alumnos diseñaron una pizarra digital de bajo costo que funciona con las netbooks del programa a través del sistema operativo Huayra. El equipo de trabajo desarrolló también un pequeño dispositivo en forma de lapicera para ser utilizado junto con la pantalla digital y una serie de juegos pedagógicos.[8] La particularidad del poyecto fue la colaboración de alumnos entusiastas desplegando sus habilidades e iniciativas para colaborar en el armado de los dispositivos. Sin duda se transforma la manera de

---

8Les compartimos el link de “Protagonistas de la Educación”, donde el Profesor Cristián Liberatore relata, en primera persona, la experiencia de trabajar con Huayra Linux https://youtu.be/_SPYyzLKBbY

aprender y enseñar y sobre todo despierta la curiosidad de muchos otros alumnos que poseen las netbooks y no conocen en su totalidad sus ventajas educativas y científicas.

Para finalizar podemos decir que el avance argentino en materia de tecnología fue muy significativo en los últimos años, pero es un camino que recién se inicia y debe profundizarse y establecerse como una política de Estado y un derecho adquirido por todos los ciudadanos, que nos encamine hacia la independencia tecnológica y la igualdad de oportunidades. aún así, no podemos dejar de reconocer que Huayra representa un antes un después en la historia educativa y la historia tecnológica de nuestro país.

Bibliografía consultada

Busaniche, Beatriz "El software Libre es esencial para la Libertad de Expresión" disponible en http://www.unr.edu.ar/noticia/298/beatriz-busaniche-el-software-libre-es-esencial-para-la-libertad-de-expresion. Última actualización 6 de enero de 2016.

Castrillo, J. (2013, enero 29) "Como se desarrolló el Linux de las Netbooks Educativas" disponible en http://www.lanacion.com.ar/1549077-como-se-desarrollo-el-linux-de-las-netbooks-educativas. Última actualización 06 de enero de 2016.

Hernández Jordi más Software Libre: técnicamente viable, económicamente sostenible y socialmente Justo Infonomia. Red de Innovadores, Barcelona 2005.

Páginas web consultadas

http://www.educ.ar/

http://www.conectarigualdad.gob.ar/

http://www.redusers.com/

http://huayra.conectarigualdad.gob.ar/

EXPERIENCIAS CON EL SOFTWARE LIBRE

Paula Betiana Arancibia

DE LA TIERRA A LA LUNA CON EL SOFTWARE LIBRE

El Software Libre se refiere a cuatro libertades de los usuarios del software: la libertad de usar el programa, con cualquier propósito; de estudiar el funcionamiento del programa, y adaptarlo a las necesidades; de distribuir copias, con lo cual se puede ayudar a otros y de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie.[9]

Las escuelas, el ámbito educativo, no es menos en cuanto a esta necesidad. Como futuros docentes tenemos que entender las estrategias que debemos utilizar en torno a las TIC, por lo que el beneficio que nos produce al poder realizar algún tipo de modificación a un programa para poder utilizarlo en diferentes áreas y con niños con diferentes niveles cognitivos es amplio.

9 https://www.gnu.org/philosophy/free-sw.es.html



Nosotros, los alumnos de 2do Año de la carrera Profesorado de Educación Primaria, del Instituto de Educación Superior N° 6001, hicimos la experiencia con el juego TierraLuna. Este juego para leer, trata de un cohete que avanza hacia la luna evitando golpear con meteoritos y asteroides. En la parte inferior del mismo aparecen textos, el cual son leídos mientras se juega; realizando así una lectura de varios minutos y de manera interactiva.

En el espacio curricular de Currículum y Organizadores Escolares del Nivel Primario, dictado por la Profesora Sofía Juárez; se nos dio como consigna la modificación de los textos del juego. En grupo de a 5 (cinco) integrantes como máximo se nos pidió sustituir los textos por breves resúmenes de un área que nos tocó al azar del Diseño Curricular de la provincia Salta; debíamos hacer la elección de un Eje y de allí un contenido. Los textos debían explicar de qué se trataba cada contenido, pasando por todos los grados o años del Nivel Primario. El juego finalizado debía ser compartido con los demás grupos para su análisis y comprensión de lo redactado.

El apoyo y guía en el juego fue dado por el Profesor Rodrigo Manresa, quién muy amable y predispuesto, nos explicó e instaló el sistema operativo Huayra GNU/Linux, en las computadoras de Conectar Igualdad donde el sistema estaba desactualizado o desinstalado.

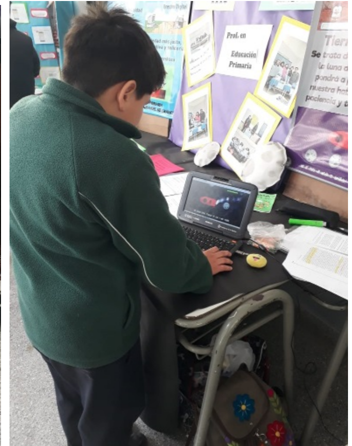
Comenzamos el proyecto el cual tuvo un resultado mejor de lo que esperábamos. No sólo por la facilidad de cómo podíamos ajustar un juego a nuestros requerimientos, sino también por la practicidad y el abanico de opciones que nos da el software libre para el momento de enseñanza-aprendizaje.



El proyecto fue presentado en la Feria de Ciencia y Tecnología 2019 instancia zonal (Esc. Dr. Benjamín Zorrilla). Los alumnos de las instituciones que visitaron los stands quedaron fascinados por el

juego; docentes de plástica y música sentían curiosidad de cómo podrían adaptar el juego a su espacio curricular, se interesaron mucho en el manejo, diseño y configuración del juego[10]. Si bien no fue el ganador de esa instancia, nos abrió las puertas para seguir mostrando el proyecto y ganar experiencia.





Participamos en varios eventos más como lo es ORA-2019 por el IITA (UCaSal). Allí varias personas involucradas en el sistema educativo estuvieron muy interesadas en el proyecto. Tanto docentes

10 https://github.com/HuayraLinux/TierraLuna

como visitantes y niños mostraban su aceptación a un diseño que se puede utilizar en las escuelas, también en la facilidad de la modificación que se puede hacer para resultar algo muy útil en varias áreas. Los niños por su parte se mostraban atraídos por el juego, pudiendo experimentar la funcionalidad del mismo.







También la disertación en Software Freedom Day 2019 (UNSa) llevada a cabo el día 3 de octubre. En ella observamos el interés de los concurrentes; las preguntas realizadas fueron muy gratificantes y motivadoras para seguir dando a conocer el objetivo del proyecto. Hubo varias sugerencias también que nos dio algunas ideas y más ganas de seguir aprendiendo y ver el alcance que se le puede dar.


TIERRALUNA
EDUCAR CON HUAYRA GNU/LINUX


Interrogantes accesorias

Es una excelente herramienta para trabajar en las aulas; es una modalidad que atrae, que gusta, entretiene y tiene un objetivo fuerte. En cuanto a los docentes que se resisten a la tecnología o aquellos que quieren innovar en sus clases este mecanismo es el correcto. No hace falta ser experto en el tema, es tan fácil como cortar y pegar. Y esto es exactamente lo que brinda el software libre. Sólo basta con tener una buena idea e intentar abordarla.



El cielo es el límite…o, en este caso, la luna.



“Estoy muy agradecida por el apoyo de mis profesores, por darme siempre oportunidades únicas y tener tanta confianza en mí. Mi familia que siempre estuvo aguantando tantos trajines propios de la carrera; sabiendo que los frutos de todo esto es también para ellos. Y más que nada a los grandes amigos que se cruzaron en mi vida. Como siempre, la perseverancia ante todo.”

EL CRECIMIENTO EN LOS VIDEOJUEGOS

Jorge Luis Ventura

Rodrigo Gastón Manresa

LOS VIDEOJUEGOS LIBRES

En este capítulo te mostraremos todos o casi todos los juegos libres que existen disponibles para el sistema operativo de Linux, aquí los vamos a analizar, destripar y detallar con precisión para ti, para que puedas decidir con acierto que juego bajar o comprar, además también para que puedas observar que en Linux existen muchos juegos y que cada vez son más y más.

Existen varias opciones a la hora de jugar en nuestra computadora con Linux, está la herramienta como **WINE** con su interfaz **PlayOnLinux** que facilita la tarea a la hora de instalar y configurar juegos de Windows y sus componentes vitales **Dx Wine, Winetricks y Lutris.** También tenemos el no tan nuevo sistema operativo Steam Os, Ozon OS, Linux Gamers Live (en sus versiones Big y Lite con juegos preinstalados), SparkyLinux y otras más.

Por último hablaremos sobre los principales títulos de videojuegos nativos para Linux.

Herramientas para instalar juegos que corren en Windows:

WINE



Varios programadores en conjunto hicieron un programa llamado WINE, pero ¿Que significa Wine is not an emulator?
Que es una de las formas de correr un programa de Windows bajo Linux. La razón por la cual Wine no es un emulador es que los emuladores tienden a duplicar el entorno completo en el que un programa vive, incluyendo la simulación de una arquitectura de microprocesador determinada. Wine, por el contrario, implementa lo que podría ser llamado una capa de compatibilidad, la cual provee alternativas a las bibliotecas de Windows.

Dx WINE

¿Cuántas veces has intentado instalar directx via wine o winetriks sin resultados positivos? pues tu problema se ha terminado, este pequeño proyecto que representa meses de trabajo lo he realizado sencillamente porque muchos programas o juegos requieren de directx para su funcionamiento, además incluye algunos añadidos a la instalación que hará que prácticamente no te falte alguna dependencia con algunos programas.

Este software no ha utilizado ningún tipo de modificación a software propietario o privativo, simplemente prepara a wine para que su instalación sea efectiva con varias dependencias o remplazos gnu, ten en cuenta que en cuanto al funcionamiento de la aceleración d3d, texturas, pixel shader depende de factores como lo es tu tarjeta de vídeo, y versiones de Wine. para más información visita el sitio de wine donde te muestran la compatibilidad y estado de cada componente.

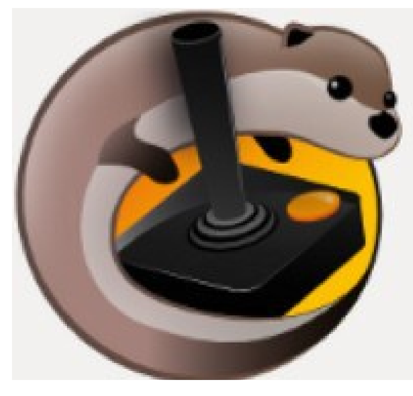
Winetricks

Se trata de un es un script que nos va a ayudar un poco en este proceso, con este script podemos descargar e instalar en wine, las principales librerías (dll's) que normalmente nos solicitan los programas que corren bajo Microsoft Windows.



Lutris

Lutris es una plataforma para la instalación sencilla de juegos en el sistema operativo del pingüino. Su sistema de funcionamiento es muy similar al Centro de Software: Buscas lo que te apetece y haces clic en "Instalar".



La **cantidad de juegos disponibles es increíble** (hay que tener en cuenta que todavía están comenzando) y podremos encontrar software libre o privativo, incluso juegos para Windows que instalaremos sencillamente con la ayuda de éste programa y Wine. También tenemos juegos para las emuladoras (Mame, NES…).

Sistemas Operativos para Gamers Linux

SteamOS

Valve ha apostado por entrar de lleno dentro del mundo de las videoconsolas. Para ello ha



desarrollado una máquina, un mando y un sistema operativo específico para dicho efecto. Este sistema operativo es SteamOS, basado en Debian, y

permite ejecutar de forma nativa los títulos específicos para Linux que ofrece la compañía a todos sus usuarios.

SteamOS se puede instalar en cualquier ordenador convencional (no necesitamos utilizar una Steam Machine para ello) y se puede descargar desde su página web.

Características

- Podrá ser instalado en cualquier PC.
- Aparentemente está más orientado a ser un media center que un SO de uso general.
- Gratuito para usuarios y desarrolladores.
- Es un sistema operativo optimizado para los juegos.
- Permitirá jugar a juegos disponibles en Steam para otras plataformas vía streaming (por ejemplo de nuestro PC a una Steam Machine en nuestro salón).

Ozon OS

Aunque pueda parecer una distro genérica y similar a otras, las cuales mimaban el diseño y el minimalismo de su interfaz, también puede verse como una buena distro para los que les gustan los videojuegos. Los desarrolladores han optimizado esta distro para juegos, agregando controladores gráficos y plataformas para videojuegos.



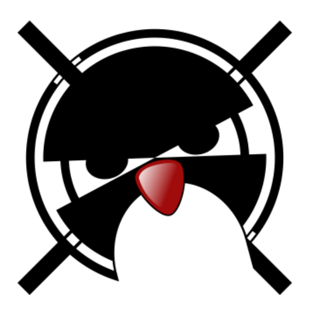
Es una distribución que nace de la colaboración entre el proyecto Numix y Nitrux S.A., dos grupos de desarrolladores que llevan años diseñando espectaculares sets de iconos y aplicaciones para Linux. Tendrá por lo tanto, **un diseño moderno y minimalista**. También nos prometen que estará optimizado para el gaming.

La primera versión alpha de Ozon fue lanzada a mediados del año pasado basándose en Fedora 20,

pero se espera que la beta y la versión final salgan durante este 2015 basándose en **Fedora 21** y utilizando el entorno **Atom Shell**, un fork de GNOME Shell que cuenta con su propio dock, panel y lanzador de aplicaciones.
De momento no se tienen características sobre este sistema que promete mucho.

Linux Gamers Live

Linux Gamers Live es una distribución Linux, pensada especialmente para gamers "libres" con gran variedad de juegos para Linux.



- La edición BIG contiene la gran selección de juegos completos. Esto incluye los juegos no aptos para niños o juegos que requieren hardware más potente. Que cabe en un DVD o una unidad flash USB (min. 5 GB de capacidad).

- La edición de Lite contiene una selección de juegos más pequeños. Sólo tiene juegos adecuados para niños y juegos que funcionan incluso en hardware antiguo. Que cabe en un CD o una unidad flash USB (mínimo 1 GB de capacidad).

Al ser “Live” no hace falta crear particiones ni instalar nada, ya que arrancará automáticamente desde la unidad de DVD, y para evitar los problemas con los drivers de las tarjetas gráficas, incluye por defecto tanto los de **NVIDIA** como los de **ATI**.

SparkyLinux

Es un sistema operativo de escritorio basado en Linux con un enfoque en los juegos de PC.



Las siguientes son las características que brinda esta distro:

- Acceso a los juegos compilados para la plataforma Linux.
- Acceso a los juegos “populares” y “modernos” a través de Steam y plataformas Desura.
- Acceso diversos juegos creados para la plataforma Windows a través de Wine y PlayOnLinux.
- Acceso a los juegos clásicos, creado para las máquinas y los sistemas a través de emuladores discontinuados.

Los siguientes emuladores pueden ser usados en esta Distro:

- DeSmuME – Emulador para los juegos de Nintendo DS.
- DOSBox – Sistema emulador DOS.
- MAME – Juegos de árcade emulador + GUI front-end GNOME.
- Nestopia – Nintendo Entertainment System emulador.

- Reloaded-PCSX – Sony PlayStation emulador.
- Stella – Emulador Atari 2600.
- Visual Boy Advance – Emulador de Game Boy, Game Boy Advance y Game Boy Color.
- Yabause – Emulador Sega Saturn.
- ZSNES – Emulador de Super Nintendo Entertainment System.

SparkyLinux ya trae todo esto ya instalado para ti. Sin duda alguna, esta distribución es la más adecuada para una sala de juegos, o para aquellos niños que les gusten los juegos, y por qué no; a aquellos informáticos que tengan alma de niño.

Sin embargo, el hecho que el enfoque de esta distro se base en los **juegos para PC** no significa que no puedes usar o hacer lo que normalmente realizas en tu trabajo o estudio mediante Calligra, Libreoffice, Microsoft Office Online (a través de un acceso directo), OpenOffice y WPS Office. Al igual que otras distribuciones, SparkyLinux, usa el último

Kernel. Además, utiliza el entorno LXDE ligero para minimizar los requisitos del sistema.

Títulos de videojuegos para GNU/Linux

0 A.D.

Es un videojuego de estrategia en tiempo real (RTS, real-time strategy) libre y de código abierto de guerra antigua. En resumen, es un juego de guerra y economía que permite a los jugadores revivir o reescribir la historia de las civilizaciones occidentales, centrándose entre los años 500 a.c y el 500 d,c. El proyecto es altamente ambicioso usando la última tecnología en gráficos 3D, ilustraciones detalladas, sonido, y un motor de juego a la medida flexible y potente.

En un principio iba a ser un “mod” para Age of Empires II: The Age of Kings, pero se dio un giro al desarrollo, creando un juego completamente independiente, basado en sus ideas.



Frozen-Bubble

Es un videojuego de puzzles inspirado en Puzzle Bobble, en el que se intenta disparar burbujas en grupos del mismo color para hacerlos estallar hasta que no quede ninguna.



Hedgewars

Es un videojuego de estrategia por turnos basado en la popular saga Worms pero con erizos rosados en lugar de gusanos. El objetivo del juego es derrotar al resto de equipos participantes usando diversas armas, herramientas y la ventaja táctica del terreno.



Nexuiz

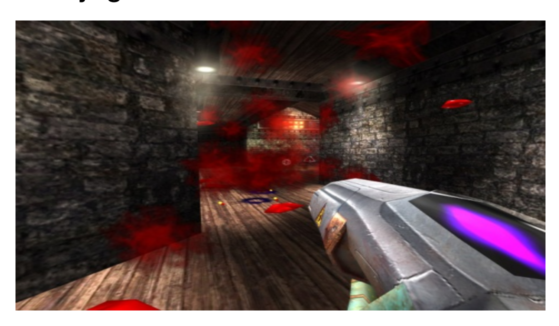
Es un juego de acción en primera persona creado por Alientrap Software. Usa DarkPlaces, que es una modificación del Motor de juego del Quake. El logo está basado en el carácter chino, ["力] que es el carácter que representa tendón, y denota fuerza. A

mi criterio es uno de los más divertidos y con excelentes gráficos.



Open Arena

Es el mismo juego que el Quake III Arena. Uno de los mejores y más adictivos shooters en primera persona de todos los tiempos, y toda una referencia para las nuevas creaciones en el género, ya que dispone de uno de los mejores y más adictivos modos multijugador.



Sauerbraten

Es otro shooter en primera persona que dispone de modo multijugador. Una curiosidad sobre este juego es que su motor gráfico es completamente original y se ha publicado bajo ZLIB de código abierto.



SuperTux2

Es un videojuego de plataformas en 2D fuertemente inspirado en Super Mario de Nintendo pero en lugar de Mario, el héroe de este juego es Tux, la mascota del kernel Linux. Es software libre.



SuperTuxKart

Es un juego libre y gratuito de carreras de coches tipo arcade en 3D, cuyo protagonista es Tux, la mascota del kernel Linux, entre otros personajes. Es una especie de clon de Mario Kart.



Warsow

Es un FPS de la vieja escuela que luce unos bonitos gráficos cell shading, que aunque no reparen en detalles se mueven de lujo, y dispone de un buen abanico de armas.



Warzone 2100

Es un juego de estrategia en tiempo real y tácticas en tiempo real, desarrollado por Pumpkin Studios y publicado por Eidos Interactive. El juego destaca por varios aspectos dentro de su género: incluye varias tecnologías de radar, se enfoca mucho en la artillería y tecnología anti-baterías, y diseño de vehículos personalizados. Inicialmente fue desarrollado para Microsoft Windows y PlayStation, posteriormente el código fuente y todos sus archivos de datos fueron liberados bajo la GNU General Public License, haciéndolo un juego libre.



Wesnoth

La batalla por Wesnoth (título original: *Battle for Wesnoth*) es un videojuego de estrategia por turnos con ambientación fantástica.

La batalla se da por el control de las aldeas, utilizando variedad de unidades que tienen ventajas y desventajas de los diferentes tipos de terrenos y de los diferentes tipos de ataques en contra. Las unidades ganan experiencia y niveles de avance, y se trasladan de un escenario a otro en una campaña.



World of Padman

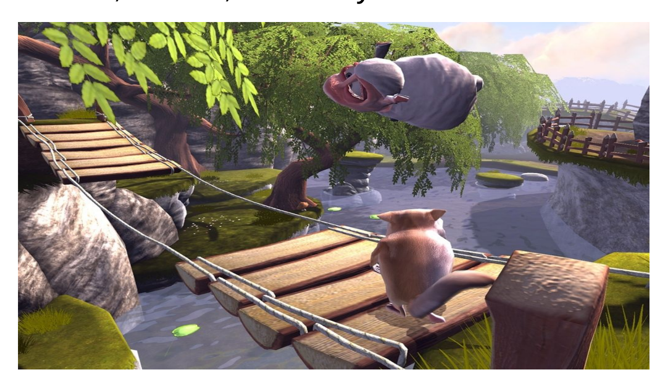
Es un juego que viene a demostrarnos que para hacer un buen shooter en primera persona no hace falta crear un ambiente oscuro y misterioso. De estilo muy desenfadado y estética de dibujos animados, este World of Padman es un juego bastante divertido y que tiene como base el motor gráfico del Quake III.



Arcades:

Existe un sinfín de títulos de juegos arcades dentro de Linux, esto se debe a que al no ser muy exigente en recursos y a la hora de correr son muy bajas las especificaciones en cuanto a software en donde se inician (java, flash, wine, etc.).

Algunos títulos rápidos: Aisleriot, Alien-Arena, Ardentryst, Armagetronad, AssaultCube, Astromenace, Blob Wars, Chromium-BSU, DreamChess, Enigma, Extreme Tux Racer, FlightGear, Freeciv, FreeDoom, Frets on Fire, FooBillard, Gweled, lbreakout2, Maniadrive, Glest, Maniadrive, Minetest, Neverball, Neverputt, OpenCity, OpenTTD, Pingus, Quadra, Quake, Red Eclipse, Teeworlds, Scorched3D, StepMania, TORCS, Tremulous, Vega Strike, WarMUX, Widelands, Xmoto, Xonotic y Yo Frankie! *



* Nota: Existen algunos juegos como AssaultCube, Blob Wars, Quake, Red Eclipse, Sauerbraten y Warsow que son una especie de híbrido ya que poseen partes libres como el motor gráfico y algunas partes privativas como el contenido multimedia.

ANDROID Y EL AVANCE DE SISTEMAS OPERATIVOS MÓVILES ABIERTOS

Marcos Alejandro López Pereyra

ANDROID, EL LÍDER EN LOS MÓVILES

¿Que es Android?

En los últimos años los teléfonos móviles han experimentado una gran evolución, desde los primeros terminales, grandes y pesados, pensados sólo para hablar por teléfono en cualquier parte, a los últimos modelos, con los que el término “medio de comunicación” se queda bastante pequeño.

Es así como nace Android. Android es un sistema operativo y una plataforma software, basado en Linux para teléfonos móviles. Además, también usan este sistema operativo las tablets, netbooks, smartwatch, reproductores de música e incluso PC’s. Android permite programar en un entorno de trabajo (framework) de Java, aplicaciones sobre una máquina virtual Dalvik (una variación de la máquina de Java con compilación en tiempo de ejecución). Además, lo que le diferencia de otros sistemas operativos, es que cualquier persona que sepa programar puede crear nuevas aplicaciones,

*widgets*1, o incluso, modificar el propio sistema operativo, dado que Android es de código libre, por lo que sabiendo programar en lenguaje Java, va a ser muy fácil comenzar a programar en esta plataforma.

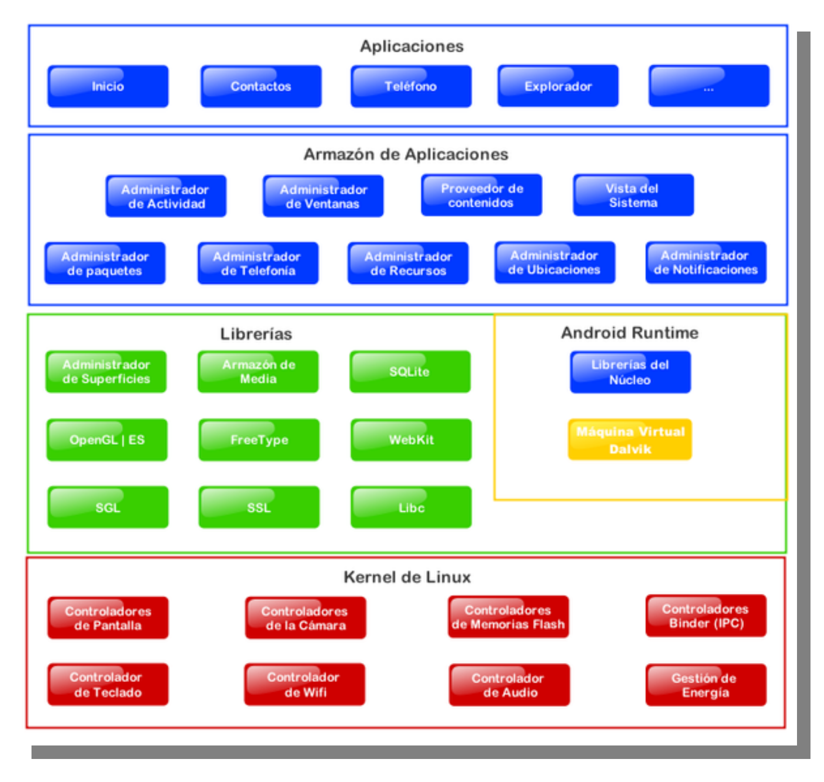
Historia de Android

Fue desarrollado por Android Inc., empresa que en 2005 fue comprada por Google, aunque no fue hasta 2008 cuando se popularizó, gracias a la unión al proyecto de Open Handset Alliance, un consorcio formado por 48 empresas de desarrollo hardware, software y telecomunicaciones, que decidieron promocionar el software libre. Pero ha sido Google quien ha publicado la mayor parte del código fuente del sistema operativo, gracias al software Apache, que es una fundación que da soporte a proyectos software de código abierto.

Dado que Android está basado en el núcleo de Linux, tiene acceso a sus recursos, pudiendo gestionarlo, gracias a que se encuentra en una

capa por encima del Kernel, accediendo así a recursos como los controladores de pantalla, cámara, memoria flash…

En la siguiente figura se muestran las capas que conforman el sistema operativo Android:



En la imagen se distinguen claramente cada una de las capas: la que forma parte del propio Kernel de

Linux, donde Android puede acceder a diferentes controladores, las librerías creadas para el desarrollo de aplicaciones Android, la siguiente capa que organiza los diferentes administradores de recursos, y por último, la capa de las aplicaciones a las que tiene acceso.

Versiones disponibles

Desde que se liberó la versión posterior al **sistema operativo Android** original, en este caso el **Android 1.0**, se han ido liberando nuevas versiones con el correr de los años, como podemos apreciar en la siguiente imagen:



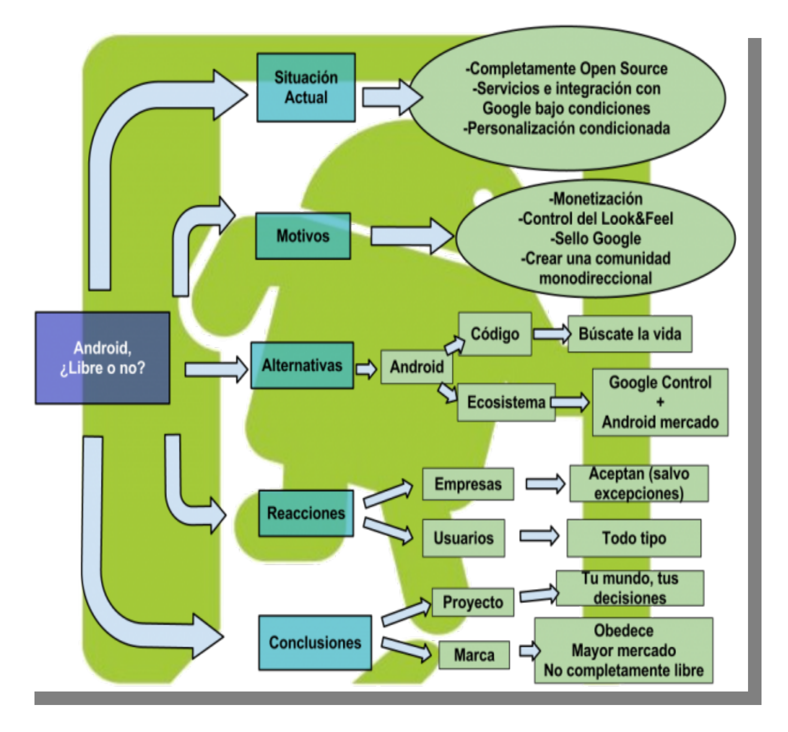
¿Android es libre o no?

La respuesta es tan sencilla como contradictoria: Sí, completamente; pero No, para nada. Me explico, si hablamos de Android como SO puramente, entonces sí, estamos hablando de un proyecto completamente libre y abierto a la comunidad para hacer con ello lo que queramos. Sin embargo, si hablamos de AndroidTM (la marca Android) y los servicios de Google que integra por defecto el SO, entonces hablamos de un ecosistema cerrado en que debe seguir unas pautas concretas para poder entrar, aunque se deje algo de libertad (sobre todo en la parte estética).

Por último me gustaría dejarles un pequeño esquema a modo de resumen condensado de las ideas expuestas, para que puedan aclarar más las dudas.

Android es un sistema operativo de código abierto basado en Linux, que principalmente se diseñó para dispositivos móviles con pantalla táctil, como

Smartphones o Tablets. Actualmente su mercado es más amplio ya que permite instalar el sistema operativo en todo tipo de dispositivos, desde Smartphones, computadoras portátiles e incluso microondas.



Razones para utilizar Android

A continuación presento las que considero son las

principales razones para utilizar este sistema operativo.

Comunidad: Android cuenta con una gran comunidad de desarrolladores y usuarios, esto se ha ido incrementando conforme pasan los años, actualmente cada día se activan más de 1 millón de dispositivos Android en todo el mundo, por lo cual nos damos cuenta del gran crecimiento en el mercado.

Aplicaciones: Android es el sistema operativo que cuenta con una mayor cantidad de aplicaciones en el mercado. Actualmente en su tienda Google Play se encuentran más de 1,000,000 de aplicaciones, tales que supera a las aplicaciones alojadas en la App Store. Otra cosa que cabe señalar dentro de las aplicaciones es que muchas de ellas, si no es que la mayoría, son gratuitas, lo cual permite a las personas acceder a ellas más fácilmente.

Código abierto: Android está liberado con licencia Apache y código abierto, lo cual permite que un

desarrollador pueda modificar su código y mejorarlo sin tener que pagar alguna licencia por ello.

Variedad de dispositivos: Android puede ser instalado en teléfonos de cualquier fabricante, esto les permite desarrollar gran variedad de dispositivos, de diferentes gamas, precios y características, sin tener que forzar a las personas a utilizar un sistema operativo o una marca en específico.



Costo: Por lo anterior es que existe una gran variedad de dispositivos móviles que usan el sistema operativo Android. La gama básica de Smartphones con Android es crucial para el éxito de este sistema operativo. No todo el mundo puede ni

quiere tener el Smartphone más caro y completo del mercado y la gama media y baja permite a muchas personas acceder a un móvil Android por poco dinero.

Personalizable: Además de ser libre, es muy personalizable, debido a que se tiene una gran cantidad de opciones que permiten modificar la apariencia de nuestro dispositivo de una manera fácil y basada en los gustos de cada persona.

Libertad: Android permite al propietario instalar lo que quiera sin poner restricciones, no importa si las aplicaciones están registradas en Google Play o son descargadas de sitios desconocidos. También le ofrece esta libertad a los desarrolladores a los cuales no les restringe el contenido de su aplicación y no es necesario que las publiquen en la Google Play, haciendo más fácil su distribución por otros sitios.

Sin fronteras: El desarrollo de Android no exclusivo de algunas operadora, permite la libertad

de que se pueda disfrutar de él siendo de la operadora que sea.

Multitarea: Android permite gestionar varias aplicaciones ejecutadas al mismo tiempo dejando en suspensión aquellas que no se utilicen y cerrarlas en caso de que ya no se vuelvan a necesitar.

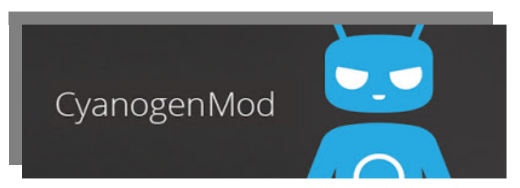
Android es de Google: Y tener detrás al gigante de la Red es algo que a muchos les estimula. Los terminales Android aseguran la compatibilidad total con los servicios de Google, tales como Gmail o Google Drive. Además, muchas funciones de estos, solo están disponibles para terminales Android y no suelen llegar a iOS o Windows Phone.

¿Qué es CyanogenOS el sistema del que todo el mundo habla?

Este sistema operativo llamado por sus desarrolladores CyanogenMod, posteriormente **CyanogenOS** fue descontinuado y sustituido en su lugar por el proyecto abierto **LineageOS**, es la

derivación de Android más famosa siendo un software libre de código abierto para smartphone y tablet basado en el sistema operativo Android.

Ofrece una mayor personalización del interfaz y promete más velocidad y una seguridad reforzada,



además de una reducción del consumo de la batería. A pesar de que no es un sistema operativo tan cerrado como el desarrollado por el gigante de Internet y presenta más flexibilidad, es compatible con todas las aplicaciones de la tienda Google Play.

En definitiva es una alternativa de Android que permite una menor dependencia de Google. No está atado a ningún fabricante y las compañías pueden modificar el código base y añadir apps**.**

Existe otro sistema llamado **Replicant** que es un clon de software libre de Android, en el que se ha tratado de eliminar componentes, bibliotecas y

Drivers privativos reemplazándolos por otros libres.

TIZEN, ABRIENDO EL FUTURO

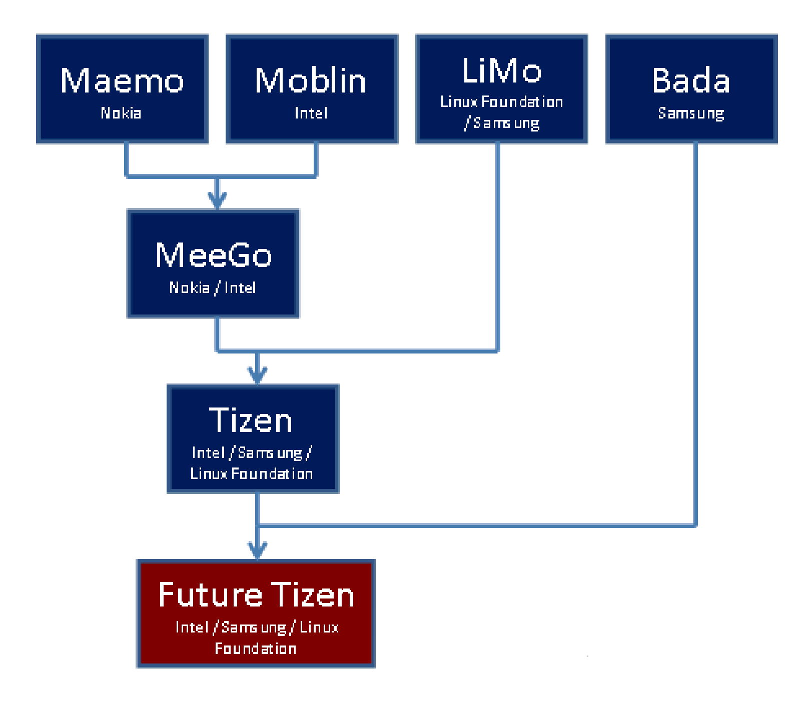
El origen

Tizen es un proyecto basado en Linux que tiene como punto de partida al malogrado **MeeGo** y que a su vez consistía en una combinación de **Moblin** (creado por Intel) y **Maemo** (obra de Nokia).

De hecho, a partir de MeeGo también ha surgido otro sistema operativo, **SailfishOS**, que ahora abandera la compañía finlandesa Jolla, creada a su vez por parte de extrabajadores de Nokia.

Actualmente, el desarrollo de Tizen está abanderado por Intel, Samsung y la Linux Foundation y cuenta con el respaldo de_empresas como eBay, Panasonic, Sharp, ZTE, McAfee, TrendMicro, Konami o Here (los mapas de Nokia), además de operadores como Sprint.

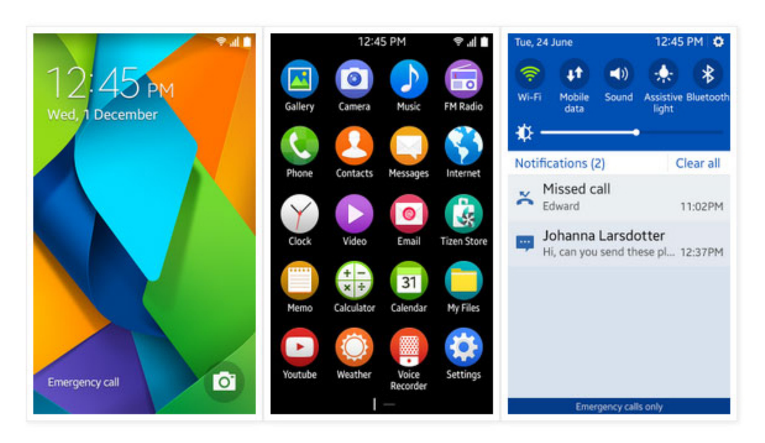
En el siguiente gráfico podemos ver más claro el árbol genealógico de estos sistemas operativos:



Próximos pasos

En concreto, las interfaces de desarrollo de Tizen están basadas en HTML5, y podemos esperar verlo tanto en smartphones como en tabletas, ordenadores, SmartTVs y otro tipo de dispositivos. No obstante, el primer aparato lanzado con Tizen en su interior fue la cámara inteligente Samsung NX300M, una versión del modelo NX300 que

estaba basado en Android.



Por otra parte, aunque Tizen nació como sistema operativo open source, con la llegada de Tizen 2 su modelo de licencias cambió ligeramente. Aunque su SDK sigue estando construido sobre código abierto, el SDK completo está publicado bajo una licencia de Samsung, pero no se espera que suponga complicaciones para otros que deseen adoptar este sistema en sus aparatos, manteniendo el espíritu original por tanto.

Samsung y Tizen

Precisamente, el papel de Samsung es una de las

grandes incógnitas que toca despejar. En teoría, se especulaba con la posibilidad de que la compañía sur coreana acabase a medio plazo reemplazando Android en sus terminales a favor de Tizen, pero tras el reciente acuerdo entre Google y el gigante sur coreano para compartir patentes y cooperar en el desarrollo de nuevos productos, puede que Tizen siga relegado a «plan b» más tiempo de lo esperado.

Habrá que ver cómo se desarrollan los acontecimientos y cómo responden también el resto de las marcas interesadas en Tizen, claro está.

Por otra parte, se espera que este nuevo sistema operativo, además de venir con aplicaciones nativas y ser capaz de funcionar con las desarrolladas mediante HTML5 (en un enfoque similar al de Firefox OS), también podrá ejecutar aplicaciones Android, eliminando de un plumazo la falta de un catálogo nutrido inicial.

FIREFOX OS, UN SISTEMA OPERATIVO PARA LOS SMARTPHONES

El Firefox OS es un sistema operativo **libre** basado en **Linux**, cuyo desarrollo se detuvo debido a la retirada de respaldo de sus socios. Su diseño está especialmente pensado para hacer correr aplicaciones en HTML5 y suele usarse en teléfonos de gama baja, como es el caso de **ZTE Open**, que fue el primer smartphone que integró este sistema operativo de serie.

La arquitectura de Firefox OS como sistema operativo se divide en tres partes básicas: **Gonk**, **Gecko** y **Gaia**. El primero de ellos es un **kernel Linux**, un sistema operativo de bajo nivel B2G. En Gecko están implementados todos los estándares y es lo que hace que se ejecute correctamente el sistema operativo así como sus aplicaciones. Por último, Gaia es la interfaz del sistema operativo.

Tras años de trabajo entre varias empresas entre las que destacan la propia Mozilla, Telefónica y

desarrolladores particulares, el Firefox OS fue anunciado en febrero de 2013 y lanzado al mercado meses después.



Para poder adquirir aplicaciones y mejoras para el sistema operativo está el Firefox Marketplace, que tiene un funcionamiento muy similar al de las plataformas de **Android** o Apple.

UBUNTU TOUCH, EL NUEVO SISTEMA OPERATIVO PARA TABLETS

Ubuntu Touch es la versión para tabletas de Ubuntu, un sistema operativo (SO) que pretende convertirse en el más utilizado por el usuario promedio que no tiene conocimientos de programación. Este Ubuntu para tabletas es la adaptación de Ubuntu Phone, para teléfonos inteligentes, que fue lanzada en enero de este año.



Por ahora Ubuntu Touch está dirigida a desarrolladores, pero los usuarios comunes pueden

“revisarla” y hacer sus aportes para mejorar esta versión, que se puede correr, por ahora, solamente en dispositivos Nexus como Galaxy Nexus, Nexus 4, Nexus 7 y Nexus 10.

Origen en Linux

Ubuntu es una "distro" (es decir, una especie de versión) de Linux, fácil de instalar; de hecho, se identifica con el lema de “Linux para seres humanos”. Es, por lo tanto, un sistema operativo de código abierto que busca convertirse en el más empleado por el usuario promedio que quiere todo hecho, sin complicaciones.

Al ser un código abierto, la versión lanzada de Ubuntu Touch puede ser descargada y mejorada por la comunidad de desarrolladores y usuarios que desde el comienzo de Ubuntu, hace siete años, ha hecho realidad más de 15 versiones evolucionadas.

Su desarrollo está patrocinado y gestionado por Canonical, una empresa británica dirigida por Mark Suttleworth, quien aseguró, durante el lanzamiento

de Ubuntu Touch, que la organización proyecta alcanzar 200 millones de usuarios para 2015, aunque considera que con las versiones para tabletas y smartphones esta meta podría alcanzarse antes.

Ventajas

- Aunque Ubuntu Touch es aún muy joven, se espera que una vez desarrollado, este SO ofrezca los mismos beneficios que su antecesor:
- Gratuito y libre: está compuesto por paquetes de software gratis y libres, esto es de código abierto en el cual programadores y aficionados conforman una comunidad amplia, junto a Canonical, haciendo cada uno aportes al código y mejorándolo.
- Es rápido y tiene como máxima la facilidad de uso y mejorar la experiencia del usuario.
- Consume pocos recursos del dispositivo.
- Su interfaz es intuitiva y tiene un diseño cuidado.
- Presume de seguro y de mantenerse libre de virus.

- Se distribuye libremente, sin costes, ni pago por actualizaciones.

El futuro

Aunque viendo sus vídeos de presentación no se puede negar que se luce muy bien, aún está por verse el futuro de Ubuntu para tabletas, sabiendo que su versión de escritorio no ha ido muy bien.

Lo tiene difícil. Competir con el iOS y Android, ya con una buena cuota de mercado, o con nuevos SO como el Windows RT, será complicado, sobre todo si tomamos en cuenta que crear todas las aplicaciones, por parte de los desarrolladores, tomará mucho tiempo.

Incluso algunos expertos en programación consideran que debió desarrollar menos su tactibilidad y hacer una transición hacia una versión sólo para desarrolladores, mientras aborda el mercado de usuarios finales, que puede ser un proceso largo.

ENCUENTROS CON EL SOFTWARE LIBRE

Ángel Miguel Giliberti

EVENTOS DE SOFTWARE LIBRE

En el transcurso del tiempo y a medida que el movimiento del software libre ha tomado relevancia en el mundo de la informática, grupos de profesionales y usuarios han creado organizaciones y actividades cuyo principal objetivo es la difusión y promoción de la filosofía, desarrollo y uso de software libre.

Las fundaciones, organizaciones y grupos aportan al movimiento y a su comunidad distintas herramientas o recursos para el pleno de desarrollo y sustento del mismo; ya sea a través del desarrollo de software libre, financiamiento o el establecimiento de reglamentos o un marco legal que los contenga. Ejemplos de estos son:

- Fundaciones u organizaciones de software libre tales como: Free Software Foundation, Digital Freedom Foundation, The Linux Foundation, Mozilla Foundation, Ubuntu Foundation, Blender Foundation.

- Grupos de Usuarios de Linux, generalmente conocidos como LUG (en inglés, Linux Users Group o GNU/Linux Users Group). Los LUG, son grupos de usuarios de GNU/Linux y software libre en general. Estos grupos son formados con la idea brindar apoyo y realizar actividades o eventos que difundan y fomenten el uso de software libre y la filosofía del libre conocimiento en sus respectivas comunidades. A continuación vamos a describir algunos de estos eventos.

FLISoL

El FLISoL (Festival Latinoamericano de Instalación de Software Libre) es el evento de difusión de software libre más grande de Latinoamérica. Se realiza desde el año 2005 y desde el 2008 el evento se realiza anualmente el cuarto sábado del mes de abril.



Imagotipo renovado del FLISoL (2015).

ORIGEN

El FLISoL nace a partir de una iniciativa por parte de Alejandro Forero Cuervo acompañado por la comunidad de Software Libre colombiana COLIBRÍ, quienes en el año 2003 organizaron un evento similar en Colombia. Este evento se conoció como FISL (Festival de Instalación de Software Libre) o FISLC (Festival de Instalación de Software Libre Colibrí).

En el año 2004 Alejandro propone mejorar la idea y durante un año y medio se sientan las bases para llevar a cabo el primer FLISoL.



FLISOL

Edit | History | Discussion

Principal: es en pt
Inscripción: es
Noticias: es
Prensa: es
Organizar: es
FAQ: es pt
Seguimiento en vivo: es
[Banners]
[Listas]
[Organizadores]
[Patrocinadores]
[Argentina]

- Capital Federal
- Casares (BsAs)
- Córdoba (Capital)
- Junín
- Lanus
- La Plata
- Mar del Plata
- Mendoza (Capital)
- Posadas (Misiones)
- Neuquén (Capital)
- Resistencia
- Rosario
- San Juan
- San Justo
- Santa Fe
- Santiago del Estero

[English | Português]

Festival Latinoamericano de Instalación de Software Libre

El **sábado 2 de abril de 2005** realizaremos el Festival Latinoamericano de Instalación de Software Libre - FLISOL, un evento cuyo propósito es promover el uso de software libre y la integración de comunidades de usuarios de software libre en todos los paises de latinoamérica.

Para lograrlo realizaremos eventos en diferentes ciudades de manera simultánea en los que los expertos instalen, de manera gratuita y totalmente legal, software libre en los computadores que para ese efecto lleven todas las personas interesadas.

Si Ud. quiere registrarse como participante del evento remitase al formulario de registro.

El Festival está siendo organizado por APESOL, BSD Venezuela, CaFeLUG, CatatumboX, CDSL, ChampeTUX, Colibrí, EcuaLUG, FUNLAM, GaLlo, GLUC, GLUD, GNU/LUGNA, GrULiC, Grupo Kernel Mexico, GUGVE, GULCR, GULIP (Puno-Perú), GULTab, GUSILA, Iskariote, LANUX, LinuxLSC, Linuxpreview, LUGRo, Merlinux, OpenEcuador, PLUG (Perú), PLUG (Argentina), POLUX, PSL-ABCD, Pulpa, Software Libre Chile, TALUG, Ulfix, UNPLUG, USMP-LUG, VELUG y XPOLINUX. Invitamos a otras comunidades de usuarios de software libre de la región a participar. **El festival se realizará en 13 paises, con un total de 106 ciudades**, 12 en Argentina, 5 en Bolivia, 30 en Brasil, 7 en Chile, 16 en Colombia, 1 en Costa Rica, 9 en Ecuador, 2 en El Salvador, 1 en Guatemala, 1 en Honduras, 11 en Perú, 4 en México y 8 en Venezuela.

Puede contactarnos escribiendo a <flisol@bachue.com> (más información sobre listas de correo).

Dentro del software que se instalará se incluyen distribuciones de GNU/Linux y BSD, así como software libre para otras plataformas como Windows y MacOS.

Fragmento de la invitación al primer FLISoL (2005).

ORGANIZACIÓN

El evento es organizado por los distintos grupos u organizaciones locales. Hasta la fecha, el FLISoL se ha realizado en países como:

Argentina, Bolivia, Brasil, Chile, Colombia, Costa Rica, Cuba, Ecuador, El Salvador, España, Guatemala, Honduras, México, Nicaragua. Panamá, Paraguay, Perú, República Dominicana,

Uruguay, Venezuela y recientemente en Alemania y España. Por la incorporación de estos últimos y el interés de otros países por formar parte del evento, se ha propuesto cambiar el nombre y significado de FLISoL por algo más inclusivo para aquellos países que no son parte de Latinoamérica.

OBJETIVOS Y ACTIVIDADES

En el FLISoL se realizan diferentes actividades, predominando la instalación, de manera gratuita y legal, de software libre en los dispositivos de los participantes que así lo deseen.



FLISoL realizado en El Carril, Salta, Argentina (2016)

En paralelo se realizan talleres, charlas, exposiciones y otras actividades para dar a conocer y estimular el uso de software libre.

Software Freedom Day

El Software Freedom Day (SFD), también conocido como "El día de la libertad del Software" es un evento, que se realiza a nivel mundial en el que se celebran las virtudes del software libre y de código abierto (Free and Open Source Software, FOSS por sus siglas en inglés) y se promueve el uso del mismo. La visión del SFD es incentivar a las personas a conectarse, crear y compartir libremente un mundo digital que sea participativo, transparente y sustentable.



Imagotipo del Software Freedom Day

ORIGEN

El origen del SDF se remonta al año 2004 con la creación de la “Software Freedom International” (desde 2013 Digital Freedom Foundation). La misma nace a partir de la idea de Matt Oquist de establecer un día internacional en el que se repartiera CDs con software libre en los foros de OpenCD. A Henrik Omma y Phil Harper, quienes eran miembro del proyecto OpenCD, les gustó la idea e inmediatamente empezaron a colaborar con la misma. Luego de considerar varias propuestas con los nombres, fechas y logística, Matt, Henrik y Phil estuvieron de acuerdo con “Software Freedom Day” ya que ellos creían que todas las personas sin un interés personal en el software propietario pueden unirse para enseñar al público mundial acerca de los ideales de la libertad del software y los beneficios prácticos del software libre. El 28 de Agosto de 2004 se realizó el primer Software Freedom Day y desde el año 2006 se realiza el

tercer sábado del mes de Septiembre.

ORGANIZACIÓN

La Digital Freedom Foundation (anteriormente llamada Software Freedom International) es la principal organizadora del evento, centraliza los canales de información oficial, brinda soporte, obsequios y puntos de colaboración; pero también existen grupos voluntarios alrededor del mundo que organizan el SFD en sus comunidades.

OBJETIVOS Y ACTIVIDADES

En el SFD se realizan diversas actividades relacionadas al software libre, tales como seminarios, charlas, demostraciones, talleres, puntos de distribución de software libre, torneos, entre otras cosas. Estas actividades se llevan a cabo con el objetivo de:

- Celebrar la libertad del software y a las personas detrás de este.
- Fomentar una comprensión general de la

libertad del software e incentivar la adopción del software libre y estándares abiertos.

- Crear un acceso más igualitario a oportunidades mediante el uso de tecnologías participativas.
- Promover un diálogo constructivo sobre los derechos y responsabilidades en la sociedad de la información.

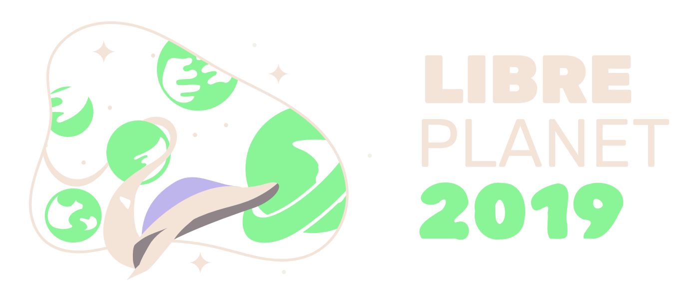


SFD realizado en la Ciudad de Salta, Argentina (2016)

Libre Planet

LibrePlanet es un proyecto comunitario creado y apoyado por la por la Free Software Foundation

(FSF). Su objetivo es la promoción del software libre alrededor del mundo, realizando cada año una conferencia internacional para las comunidades locales y organizaciones.



Imagotipo de LibrePlanet 2019.

ORIGEN

El proyecto LibrePlanet nació en 2006, en una reunión de los miembros asociados a la fundación con la voluntad de organizarse en grupos geográficos. La primera conferencia de LibrePlanet se realizó durante los días 21 y 22 de Marzo del año 2009.

ORGANIZACIÓN

El evento se realiza alrededor de los días 19 y 20

de Marzo y es organizada anualmente por la Free Software Foundation en la ciudad de Boston, Massachusetts, Estados Unidos o en sus alrededores, y está conformada por una mezcla de miembros de la fundación y voluntarios de la comunidad. La conferencia reemplaza e incorpora la Reunión Anual de Miembros (Annual Members Meeting, AMM por sus siglas en inglés) de la FSF la cual se realizaba aproximadamente al mismo tiempo cada año.

OBJETIVOS Y ACTIVIDADES

LibrePlanet establece una declaración de misión, motivación y sus respectivos objetivos.

Declaración de misión

"La misión de LibrePlanet es promover una red global de grupos locales y académicos, equipos de proyecto, grupos de trabajo basados en intereses y personas que trabajan juntas en conjunto para promover el software libre y la cultura libre como movimiento por la justicia social."

Motivación

“A medida que los medios y la tecnología median en cómo experimentamos e interactuamos con el mundo, los propietarios privados de software, conocimiento y cultura tienen control sobre nosotros. Con un fuerte compromiso con el software libre y las obras culturales libres, trabajamos contra la explotación, el dominio y la opresión permitidos por la propiedad privada de la tecnología, los medios y la comunicación. Si bien el software libre y la cultura libre son posturas radicales contra una forma de opresión, reconocemos que existen estructuras de control sistémicas integradas en nuestra sociedad que se infiltran en nuestro movimiento. Por lo tanto, mantenemos un Código de Conducta que los toma en cuenta.

Nuestra visión es liberar al mundo de las barreras tecnológicas y legales para que todo el software y las obras culturales sean libres, según lo definido

por el Proyecto GNU y la Definición de Obras Culturales Libres.”

Objetivos

- **Unir:** Crear conexiones entre hackers, activistas y participantes en el software libre y en el movimiento de la cultura libre.
- ***Edu*car:** Producir y distribuir recursos informativos sobre software libre y cultura libre.
- **Promover:** Proveer soporte y orientación para las personas que usan y contribuyen al software libre.
- **Conectar:** Formar alianzas con organizaciones relacionadas y formar coaliciones con otros grupos que trabajan contra la opresión.
- **Fomentar:** Fomentar el desarrollo de software libre y obras culturales libres.

En lo que respecta a las actividades, la conferencia provee una oportunidad para que los activistas de

la comunidad, expertos en la materia y personas buscando soluciones puedan reunirse y discutir los problemas actuales en tecnología y ética.



Richard Stallman pronuncia su discurso de apertura en LibrePlanet 2019.

Jornadas de Software Libre

Las Jornadas de Software Libre son un evento regional, internacional e itinerante donde diferentes miembros de la comunidad trabajan para integrar proyectos, proponer nuevas ideas y llevar al máximo potencial los programas que utilizan. Este

evento se realiza en varios países de Latinoamérica.

ORIGEN

La primera edición de las JSL (Jornadas de Software Libre) se realizó entre los días 4 y 6 de Agosto del año 2000 en el Centro Cultural Bernardino Rivadavia, Rosario, Santa Fe, Argentina. Asistieron alrededor de 1500 personas, muchas de ellas del interior del país y Buenos Aires. También participaron personas de otros países sudamericanos como: Brasil, Perú, Uruguay y Venezuela.

Las jornadas fueron organizadas por el LUGRo: “GNU/Linux User Group Rosario” (hoy denominado LUGRo: Grupo de Usuarios de Software Libre de Rosario) y la Asociación de Nuevas Tecnologías. En este evento se contó con la visita de Richard M. Stallman por primera vez en América Latina. Estas jornadas eran las primeras donde se hablaba del Software Libre como suceso más

representativo, y fueron rápidamente difundidas como una de las mejores en su tipo en el cono sur por la diversidad de charlas (29 charlas en 3 días).



Afiche de las primeras jornadas en Salta, Argentina (2006)

ORGANIZACIÓN

Las JSL por lo general son organizadas por los LUG acompañados o auspiciados por fundaciones, organizaciones, asociaciones, institutos educativos e incluso por el gobierno.

OBJETIVOS Y ACTIVIDADES

Durante las jornadas se reúnen programadores, desarrolladores, expertos en tecnologías, emprendedores involucrados en software libre y cualquier persona interesada con el objetivo de realizar talleres, cursos, charlas, intercambiar ideas, compartir técnicas, discutir y explorar tecnologías libres.



Jornadas de Software Libre en Montevideo, Uruguay (2002)

LIBRECON

LIBRECON es un evento diferente en comparación

a los anteriores. Está orientado a la parte comercial del software libre, y es toda una referencia del sur de Europa en el sector de las tecnologías abiertas aplicadas a sectores estratégicos de la economía.

LIBRECON

powered by CEBIT

Logotipo de LIBRECON (modificado, colores invertidos)

ORIGEN

Los orígenes de LIBRECON pueden resultar un poco confusos, así que vamos a ir por partes:

- Málaga, España, año 2004. Se realiza la Open Source World Conference (OSWC), una conferencia internacional orientada principalmente a las empresas que desarrollaban y usaban software libre.
- Mérida, España, 2005. Se realiza la OSWC.
- Málaga, 2006. Se realiza una nueva edición

de la OSWC.

- Badajoz, España, 2007. La OSWC pasa a llamarse Free Software World Conference (FSWC).
- Málaga, 2008. Se realiza otra edición con su antiguo nombre, OSWC.
- Cáceres, España, 2009. El evento se realiza, en esta ocasión, como FSWC.
- Málaga, 2010. Una semana antes del evento, la organización de la OSWC comunicó que esta había sido cancelada por problemas de presupuesto. Entonces un grupo local organizó una conferencia alternativa durante las mismas fechas y en el mismo sitio y la llamó Libre Software World Conference (LSWC), bajo el lema *"Nosotros no cancelamos tu libertad"*.
- Zaragoza, España, 2011. Se realiza una nueva edición de la LSWC.
- Santiago de Compostela, España, 2012 y

2013. Se realiza la LSWC con total normalidad.

- Granada, España, 2012. Luego de dos años de aquel evento cancelado en el 2010, se realiza por última vez la OSWC.
- Bilbao, España, 2014. La LSWC pasa a llamarse LIBRECON.

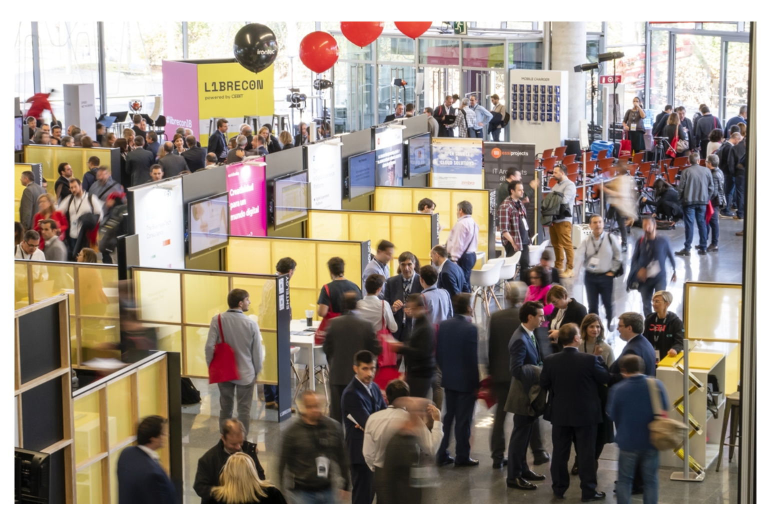
ORGANIZACIÓN

El evento es organizado por ESLE, Asociación de Empresas de Tecnologías Libres y Conocimiento Abierto de Euskadi, y Deutsche Messe, entidad alemana de organización de congresos y ferias profesionales.

OBJETIVOS Y ACTIVIDADES

El principal objetivo es la creación de negocios y empleos en todos los sectores de la sociedad, a través de las tecnologías libres, la innovación y el emprendimiento. Otro objetivo es demostrar cómo las tecnologías abiertas en general y el software

libre en particular, son una alternativa real de éxito que ofrece a empresas y administraciones públicas una mayor competitividad, grandes ahorros y desarrollo de economías locales.



Stands ubicados en la zona Expo. LIBRECON (2018)

En el evento se reúnen profesionales de las open technologies en campos como Internet of Things (IoT), Máquinas Inteligentes, Movilidad Futura, Transformación Digital y Ciberseguridad y realizan ponencias, charlas técnicas y workshops con espacios dedicados a generar ocasiones de

negocio. Por eso mismo el evento está distribuido en espacios diferenciales diseñados para la mejor convivencia de sus diferentes facetas.

OpenExpo

De manera similar a LIBRECON, OpenExpo es un evento orientado al sector empresarial. OpenExpo es una feria y congreso sobre open source que se realiza anualmente en Madrid, España.



Imagotipo de OpenExpo

ORIGEN

El primer OpenExpo se realizó, de la mano de Philippe Lardy y Manu Sánchez, el 4 de Octubre de 2012 y su eje principal fue el ecommerce (comercio electrónico).

OBJETIVOS Y ACTIVIDADES

El objetivo de OpenExpo es el de presentar, difundir, fomentar y evaluar las soluciones que ofrece el open source y el software libre en la industria, así como ayudar a las empresas y profesionales a conocer estas. Algo que destaca a OpenExpo es que es un punto de encuentro para empresas, empresarios y expertos en el tema para que estos puedan informarse sobre las últimas tendencias, aumentar la red de contactos, crear y acceder a oportunidades de empleos y negocios.



Sector de exposiciones. OpenExpo (2017)

El Congreso tiene muchas áreas y actividades diferentes que abarcan sectores como IoT, Inteligencia Artificial, Machine Learning, Cloud, Ciberseguridad, DevOps, entre otros. Cada sector cuenta con conferencias y charlas, talleres, actividades de reclutamiento, espacios para nuevas empresas, debates abiertos y más.

Por otro lado, en la tercera edición de OpenExpo (2016), se inauguraron los Open Awards. Los Open Awards tienen la intención de reconocer a las empresas y administraciones que crean y fomentan grandes soluciones con tecnologías Open Source. Los objetivos de Open Awards son:

- Reconocer y premiar los proyectos que más han destacado durante el último año.
- Impulsar la comunicación y la notoriedad pública de las empresas, administraciones y medios participantes en los premios.
- Valorar el trabajo realizado por las empresas, medios y administraciones públicas.

PROGRAMANDO HACIA LA LIBERTAD

Ángel Yael Edgardo Moya

PROGRAMAS Y LENGUAJES PARA PROGRAMACIÓN LIBRE

Python



Python es un lenguaje de programación libre preparado para realizar cualquier tipo de programa, desde aplicaciones, pasando por juegos e incluso, páginas web.

El creador del lenguaje es un europeo llamado Guido Van Rossum y su desarrollo duró varios años.

Las características más sobresalientes son:

- ➢ Permite crear todo tipo de programas y paginas web.
- ➢ Es compatible con diversos sistemas operativos siempre y cuando exista un intérprete para él.
- ➢ El código no debe ser codificado antes de su ejecución.

- Trabaja por lineas de sentencia.

Algunas empresas que lo utilizan son: Yahoo, Google, Walt Disney, la NASA, Red Hat, etc.

C

Lenguaje de programación C. También conocido como “Lenguaje de programación de sistemas” desarrollado en el año 1972 por Dennis Ritchie para UNIX un sistema operativo multiplataforma. El lenguaje C es del tipo lenguaje estructurado Sus instrucciones son muy parecidas a otros lenguajes incluyendo sentencias como if, else, for, do y while... . Aunque C es un lenguaje de alto nivel (puesto que es estructurado y posee sentencias y funciones que simplifican su funcionamiento) tenemos la posibilidad de programar a bajo nivel (como en el Assembler tocando los registros, memoria, etc.). Para simplificar el funcionamiento de el lenguaje C tiene

incluidas librerías de funciones que pueden ser incluidas haciendo referencia la librería que las incluye, es decir que si queremos usar una función para borrar la pantalla tendremos que incluir en nuestro programa la librería que tiene la función para borrar la pantalla.

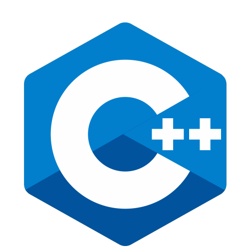
Algunas características de este lenguaje:

- Núcleo del lenguaje simple, con funcionalidades añadidas importantes, como funciones matemáticas y de gestión de archivos, proporcionadas por bibliotecas.
- Es un lenguaje estructurado, i.e. tiene estructuras de control y tipos de datos estructurados definidos por el programador a partir de los tipos atómicos típicos y mediante arreglos, estructuras, uniones y apuntadores, incluidos los apuntadores a función.
- Manejo de Interrupciones mediante la biblioteca signal.
- Usa un lenguaje de preprocesado, el

preprocesador de C, para tareas como definir macros e incluir múltiples archivos de código fuente.

➢ Se diseñó como un lenguaje de programación de sistemas portátil.

C++



C++ es un lenguaje de programación diseñado en 1979 por Bjarne Stroustrup. La intención de su creación fue extender al lenguaje de programación C mecanismos que permiten la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

Posteriormente se añadieron facilidades de programación genérica, que se sumaron a los paradigmas de programación estructurada y programación orientada a objetos. Por esto se

suele decir que el C++ es un lenguaje de programación multiparadigma.

Actualmente existe un estándar, denominado ISO C++, al que se han adherido la mayoría de los fabricantes de compiladores más modernos. Existen también algunos intérpretes, tales como ROOT.

El nombre "C++" fue propuesto por Rick Mascitti en el año 1983, cuando el lenguaje fue utilizado por primera vez fuera de un laboratorio científico. Antes se había usado el nombre "C con clases". En C++, la expresión "C++" significa "incremento de C" y se refiere a que C++ es una extensión de C.

las características más comunes son:

➢ Su sintaxis es heredada del lenguaje C.

➢ su paradigma es Programa orientado a objetos (POO).

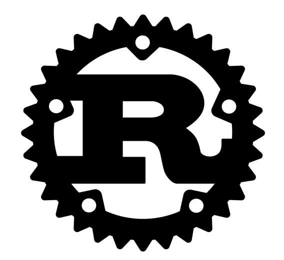
➢ Permite la agrupación de instrucciones

➢ Lenguaje muy didáctico, con este lenguaje puedes aprender muchos otros lenguajes con gran

facilidad.

- Permite la separación de un programa en módulos que admiten compilación independiente.

Rust



Rust es un lenguaje de programación desarrollado por los ingenieros de mozilla, está diseñado para desarrollar software de sistemas, donde la interacción con el usuario es prácticamente nula; excelente para aplicaciones con el modelo *cliente-servidor* o para cosas más internas en nuestra empresa. El lenguaje surgió de un proyecto personal desarrollado por Graydon Hoare (trabajador de Mozilla), quien empezó a trabajar en él en 2006; Mozilla se involucró en este proyecto en 2009 y lo dio a conocer oficialmente en 2010. Ese mismo año, el trabajo pasó del compilador inicial (escrito en OCaml) al compilador autocontenido,

escrito en sí mismo.

El objetivo de Rust es ser un buen lenguaje para la creación de grandes programas del lado del cliente y del servidor que se ejecuten en Internet. Esto ha llevado a un conjunto de características con un énfasis en la seguridad, el control de distribución de la memoria y la concurrencia. Se espera que el rendimiento de código seguro sea más lento que C++, si el rendimiento es la única consideración, pero si lo comparamos con el código C++ hecho para que tome precauciones comparables a las que toma Rust, este último puede ser incluso más rápido. La sintaxis de Rust es similar a la de C y C++, con bloques de código delimitados por llaves y estructuras de control de flujo tales comoif,else,do,while yfor. No todas las estructuras de C y C++ están presentes, además, otras (como la palabra clavematch para ramificación multidireccional) serán menos familiares para programadores que vienen de estos lenguajes.

Algunas características:

- El rico sistema de tipos de Rust y su modelo de propiedad (ownership) garantizan seguridad de memoria y seguridad de hilos, y te permiten eliminar muchas clases de bugs en tiempo de compilación.
- Rust tiene una documentación genial, un compilador accesible con mensajes de error útiles, y herramientas de primera: gestor de paquetes y de proyecto integrado, soporte avanzado multi-editor con auto completado e inspecciones de tipos, auto-formateador, etc.
- Al igual que con Python, Rust cuenta con su propio manejador de paquetes; Este manejador tiene por nombre Cargo, con él podemos crear nuestros propios proyectos, utilizar librerías de terceros, así cómo crear y dar de altas nuestras propias librerías.
- Posee una gestión automática de guardado así como una compilación nativa y estática.

GO!



Go es un lenguaje de programación creado en el año 2007 por Google. Inicialmente pretendía ser un estándar interno de codificación para la propia infraestructura de la empresa, pero su potencial lo impulsó a salir a Bolsa. En 2009 se lanza y ha ido creciendo exponencialmente desde entonces.

Sus diseñadores iniciales fueron Robert Griesemer, Rob Pike y Ken Thompson. Go es un lenguaje de programación compilado, concurrente, imperativo, estructurado, orientado a objetos y con recolector de basura que de momento es soportado en Windows y diferentes tipos de sistemas UNIX, incluidos GNU/Linux, FreeBSD, Mac OS X y Plan 9 (puesto que parte del compilador está basado en un trabajo previo sobre el sistema operativo Inferno). Las arquitecturas soportadas son i386, amd64 y ARM. Si hay algo en lo que Go sin duda destaca es

en el tema de concurrencia. En Go no existen los Threads perse, en lugar de ellos existen las goroutine, que son algo así como Threads ligeros. Con los goroutine y los canales podemos crear cientos de miles de "hilos" sin ningún problema. Esto hace a Go una excelente opción si lo que deseamos es optimizar la mayor cantidad de recursos posibles. Existen decenas de casos en los cuales empresas han dejado de utilizar X tecnología para comenzar a usar Go. Uno de los casos es el de Iron.io, en su artículo nos cuentan como pasaron de tener 30 servidores con Ruby on Rails a tener unicamente 2 con Go.

Durante muchos años, los desarrolladores han tenido que elegir entre lenguajes de programación de tipado estático compilados, por regla general, bastante complejos pero que proveen grandes características relacionadas con la optimización y el rendimiento y lenguajes de tipado dinámico interpretados con características de mucho más

alto nivel que hacían su aprendizaje, uso y sintaxis más sencilla y por tanto divertido programar en ellos, eso si, sacrificando rendimiento y control.

Go mezcla lo mejor de ambos mundos y nos aporta una sintaxis sencilla, fácil de interpretar y divertida junto a la potencia que nos ofrece un lenguaje fuertemente tipado y compilado incorporando además características de alto nivel que facilitan el uso del lenguaje por los desarrolladores.

Algunas características:

- La concurrencia es una característica extremadamente importante para un lenguaje de programación y, además, es una de las mejores cualidades de Go. La simultaneidad se relaciona con la capacidad de ejecutar más de una tarea al mismo tiempo. Go fue creado para soportar eso con sus Goroutines: pueden ejecutarse miles a la vez. Los canales les permiten sincronizarse entre sí. Así, son mucho más eficientes que

los hilos, los cuales podrían descontrolarse.

- Una de las razones por las que se desarrolló Go fue para conseguir que la programación fuera menos difícil. Es más, el código escrito en Go es limpio y de sencillo mantenimiento.
- Go ha demostrado ser una potente herramienta para las grandes empresas. Firmas como Netflix, The Economist, The New York Times, IBM y GitHub la utilizan como parte de su infraestructura interna.
- Go es un proyecto opensource, usa una sintaxis similar a C, Aun siendo un lenguaje diseñado para la programación de sistemas, provee de un recolector de basura, reflexión y otras capacidades de alto nivel que lo convierten en un lenguaje muy potente.
- Go usa tipado estático (statically typed) y su rendimiento es comparable al de lenguajes como C y C++ ya que, al igual que estos, el compilador convierte el código de Go a

código máquina.

- Go tiene muchas de las características y facilidad de lenguajes dinámicos como Python.

Java



Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta super computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes, que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir de 2012, uno de los lenguajes de programación más

populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos diez millones de usuarios reportados.

El lenguaje de programación Java fue originalmente desarrollado por James Gosling, de Sun Microsystems (constituida en 1983 y posteriormente adquirida el 27 de enero de 2010 por la compañía Oracle), y publicado en 1995 como un componente fundamental de la plataforma Java de Sun Microsystems. Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son compiladas a bytecode (clase Java), que puede ejecutarse en cualquier máquina virtual Java (JVM) sin importar la arquitectura de la computadora subyacente.

La compañía Sun desarrolló la implementación de referencia original para los compiladores de Java, máquinas virtuales y librerías de clases en 1991, y las publicó por primera vez en 1995. A partir de

mayo de 2007, en cumplimiento de las especificaciones del Proceso de la Comunidad Java, Sun volvió a licenciar la mayoría de sus tecnologías de Java bajo la Licencia Pública General de GNU. Otros también han desarrollado implementaciones alternas a estas tecnologías de Sun, tales como el Compilador de Java de GNU y el GNU Classpath.

Algunas características:

- Lenguaje totalmente orientado a Objetos. Todos los conceptos en los que se apoya esta técnica, encapsulación, herencia, polimorfismo, etc., están presentes en Java.
- Lenguaje simple. Java posee una curva de aprendizaje muy rápida. Resulta relativamente sencillo escribir applets interesantes desde el principio. Todos aquellos familiarizados con C++ encontrarán que Java es más sencillo, ya que se han eliminado ciertas características, como los punteros. Debido a su semejanza con C y C++, y

dado que la mayoría de la gente los conoce aunque sea de forma elemental, resulta muy fácil aprender Java. Los programadores experimentados en C++ pueden migrar muy rápidamente a Java y ser productivos en poco tiempo.

➢ Robusto. Java fue diseñado para crear software altamente fiable. Para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución. Sus características de memoria liberan a los programadores de una familia entera de errores (la aritmética de punteros), ya que se ha prescindido por completo de los punteros, y la recolección de basura elimina la necesidad de liberación explícita de memoria.

➢ Dinámico. El lenguaje Java y su sistema de ejecución en tiempo real son dinámicos en la fase de enlazado. Las clases sólo se enlazan a medida que son necesitadas. Se pueden enlazar nuevos módulos de código bajo demanda, procedente de fuentes muy variadas, incluso desde la Red.

Groovy



Groovy es un lenguaje de programación orientado a objetos implementado sobre la plataforma Java. Tiene características similares a Python, Ruby, Perl y Smalltalk. La especificación JSR 241 se encarga de su estandarización para una futura inclusión como componente oficial de la plataforma Java.

Groovy usa una sintaxis muy parecida a Java, comparte el mismo modelo de objetos, de hilos y de seguridad. Desde Groovy se puede acceder directamente a todas las API existentes en Java. El bytecode generado en el proceso de compilación es totalmente compatible con el generado por el lenguaje Java para la Java Virtual Machine (JVM), por tanto puede usarse directamente en cualquier aplicación Java. Todo lo anterior unido a que la mayor parte de código escrito en Java es totalmente válido en Groovy hacen que este

lenguaje sea de muy fácil adopción para programadores Java; la curva de aprendizaje se reduce mucho en comparación con otros lenguajes que generan bytecode para la JVM, tales como Jython o JRuby. Groovy puede usarse también de manera dinámica como un lenguaje de scripting.

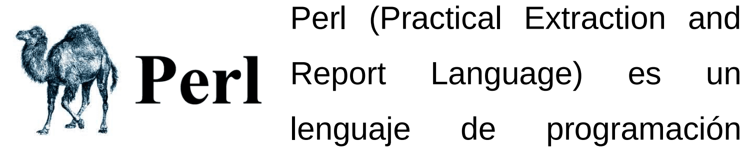
Groovy 1.0 apareció el 2 de enero de 2007. Después de varias versiones beta y otras tantas candidatas a release, el 7 de diciembre de 2007 apareció la versión Groovy 1.1 que finalmente fue renombrada a Groovy 1.5 con el fin de notar la gran cantidad de cambios que ha sufrido con respecto a la versión 1.0. En diciembre de 2009 se publicó la versión 1.7.

sus características más esenciales son:

- El lenguaje Groovy es un superconjunto del lenguaje Java. En general se puede renombrar un archivo .java en .groovy y va a funcionar
- incluyen el tipado estático y dinámico, closures, sobrecarga de operadores, sintáxis nativa

para la manipulación de listas y maps,soporte nativo para expresiones regulares, iteración polimórfica, expresiones embebidas dentro de strings.

Perl



Perl (Practical Extraction and Report Language) es un lenguaje de programación desarrollado a finales de los años 80 por Larry Wall a partir otras herramientas de UNIX como son: ed,grep,awk,c-shell, para la administración de tareas propias de sistemas UNIX.

Es un lenguaje de script de tipo BCPL (como TCL o PHP), muy semejante al AWK (de hecho está basado en él), de tipo estructurado con trazas de orientación a objetos (no completamente soportado de forma directa), que permite el desarrollo rápido de aplicaciones y herramientas especialmente orientadas al tratamiento de textos y archivos,

aunque actualmente también se utiliza incluso para entornos graficos, en combinación con sistemas como Perl/TK o GTK.

Sus características:

➢ La estructura completa de Perl deriva ampliamente del lenguaje C. Perl es un lenguaje imperativo, con variables, expresiones, asignaciones, bloques de código delimitados por llaves, estructuras de control y subrutinas.

➢ toma características de la programación shell. Todas las variables son marcadas con un Sigilo precedente (Sigil, en inglés). Los sigilos identifican inequívocamente los nombres de las variables, permitiendo a Perl tener una rica sintaxis. Notablemente, los sigilos permiten interpolar variables directamente dentro de las cadenas de caracteres (string). Como en los shell, Perl tiene muchas funciones integradas para tareas comunes y para acceder a los recursos del sistema.

➢ Perl toma las listas del Lisp, hash (memoria

asociativa) del AWK y expresiones regulares del sed. Todo esto simplifica y facilita todas las formas del análisis sintáctico, manejo de texto y tareas de gestión de datos.

➢ Todas las versiones de Perl hacen el tipificado automático de datos y la gestión de memoria. El intérprete conoce el tipo y requerimientos de almacenamiento de cada objeto en el programa; reserva y libera espacio para ellos según sea necesario. Las conversiones legales de tipo se hacen de forma automática en tiempo de ejecución; las conversiones ilegales son consideradas errores fatales.

Ruby



Ruby es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro Matz Matsumoto, quien comenzó a trabajar en Ruby en

1993, y lo presentó públicamente en 1995. Combina una sintaxis inspirada en Python, Perl con características de programación orientada a objetos similares a Smalltalk. Comparte también funcionalidad con otros lenguajes de programación como Lisp, Lua, Dylan y CLU. Ruby es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre.

Su creador ha comentado de él:

“Ruby es simple en apariencia, pero complejo por dentro, como el cuerpo humano.”

las características más destacables de este lenguaje son:

- ➢ expresiones regulares nativas similares a las de Perl a nivel del lenguaje
- ➢ Hilos de ejecución simultáneos en todas las plataformas usando green threads
- ➢ Carga dinámica de DLL/bibliotecas compartidas en la mayoría de las plataformas

- ➢ soporta alteración de objetos en tiempo de ejecución
- ➢ cuatro niveles de ámbito de variable: global, clase, instancia y local.

PHP

PHP, acrónimo recursivo en inglés de PHP: Hypertext Preprocessor (preprocesador de hipertexto), es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el preprocesado de texto plano en UTF-8. Posteriormente se aplicó al desarrollo web de contenido dinámico, dando un paso evolutivo en el concepto de aplicación en línea, por su carácter de servicio.

Su implementación en los documentos HTML era aparentemente muy sencilla. Hay que decir, que, PHP no genera HTML, sino que ofrece una salida de texto con codificación UTF-8 compatible con los

documentos HTML. El programador puede dotar a la salida de los tag's propios del HTML y los exploradores más comunes para navegar por internet, reconocerán muy rápidamente el formato UTF-8 y lo adaptarán ofreciendo una salida entendible.

Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en un documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera el texto plano en formato UTF-8, ampliamente reconocido por el estándar HTML, dando como resultado, en los exploradores, una salida al usuario perfectamente entendible.

Fue creado originalmente por Rasmus Lerdorf en el año 1995. Actualmente el lenguaje sigue siendo desarrollado con nuevas funciones por el grupo PHP.2 Este lenguaje forma parte del software libre

publicado bajo la licencia PHPv3_01, una licencia Open Source validada por Open Source Initiative. La licencia de PHP es del estilo de licencias BSD, sin la condición de copyleft asociada con la Licencia Pública General de GNU.

Sus características:

➢ Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.

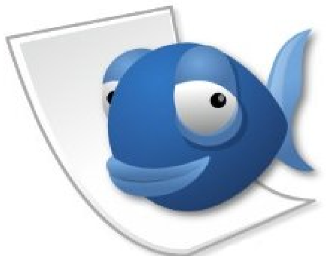
➢ El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador.

➢ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.

➢ No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.

➢ Tiene manejo de excepciones (desde PHP5).

Bluefish



Bluefish es un potente editor, dirigido a programadores y desarrolladores web, con muchas opciones para escribir páginas web y código de programación usando principalmente el lenguaje de marcado **HTML**.

Soporta muchos lenguajes de programación, ya que es un proyecto de código abierto, liberado bajo la licencia GNU GPL.

También es multiplataforma, ya que funciona en la mayoría de los sistemas operativos.

Algunas de las características de Bluefish son:

- Es un editor gráfico.
- Carga cientos de archivos en cuestión de segundos.
- Interfaz para múltiples documentos.
- Sin límite en la funcionalidad deshacer/rehacer.

- Recuperación automática después de un accidente o una caída del sistema.

En síntesis Bluefish es un potente editor destinado a programadores y desarrolladores web, que ofrece numerosas opciones para crear páginas sencilla, y que soporta los lenguajes de programación web más populares.

Turtleart (Bajo el lenguaje LOGO)



Turtle Art es un entorno de programación gráfico basado en el lenguaje LOGO, en el que se pueden hacer pequeños programas y realizar diseños con una tortuga, realizando formas y diseños en la pantalla del ordenador, es decir, permite la realización de diseños basados en la programación de los desplazamientos de la tortuga.

LOGO es un lenguaje de programación, sin embargo, y más importante, es un lenguaje para

aprender. Es una herramienta útil para enseñar el proceso de aprendizaje y pensamiento. Es fácil de usar y muy intuitivo llegando a ser muy divertido y educativo para los niños que lo utilicen. Fue diseñado con fines didácticos por Danny Bobrow, Wally Feurzeig y Seymour Papert.

Sus principales características:

- Logo es fácilmente comprendido; podemos relacionarnos con la tortuga y usarla como un objeto con el cual pensar.
- A Logo se le pueden enseñar nuevos comandos y otros comandos pueden construir sobre aquellos.
- Logo es usado con preescolares como con estudiantes de matemáticas avanzadas.
- Logo es un lenguaje de programación que provee todas la herramientas necesarias para crear programas de cualquier nivel de sofisticación. Por lo que es un lenguaje que incentiva que los estudiantes exploren, que aprendan y que piensen.

Scratch



Scratch es un lenguaje de programación que facilita crear historias interactivas, juegos y animaciones y compartir sus creaciones con otras personas en la Web. Permite a los principiantes o personas sin sentido de la programación, obtener resultados sin tener que aprender a escribir de manera sintácticamente correcta primero. Scratch está escrito en Squeak (una implementación libre de Smalltalk-80), a partir de la versión 2.0 el código es reescrito en actionscript (Adobe Flash). Es un entorno de programación que facilita el aprendizaje autónomo.

Sus características son:

- Los programas pueden ser ejecutados directamente sobre el navegador de Internet.
- Este programa está basado en bloques gráficos y la interfaz que tiene es muy sencilla e intuitiva.

- A medida que los estudiantes crean programas, aprenden conceptos fundamentales de computación tales como, control de flujo, iteración (repetición o ciclos), condicionales, procedimientos, hilos múltiples y eventos.
- Permite controlar y mezclar diferentes medios (gráficas, texto, música y sonido).
- Los bloques autoencajables están categorizados por colores de acuerdo con su funcionalidad (Azul=Movimiento; Violeta=Apariencia; Fuscia=Sonido; Verde=Lápiz; Naranja=Control; Azul claro=Sensores; Verde claro=Números; Rojo=variables). Este código visual también ayuda a los estudiantes.
- En lugar de escribir instrucciones (código), permite programar con el ratón (mouse), arrastrando bloques autoencajables y soltándolos en el área de programas. Estos se comportan como piezas de Lego o de un rompecabezas, pero solo encajan si son sintácticamente correctos.

Squeak



Squeak, como la reedición de los proyectos Dynabook y Smalltalk de los años 1970, busca ser un meta-medio. Un lugar donde convivan todos los medios de expresión conocidos hasta la fecha (texto, video, sonido, música, gráficos 2D, gráficos 3D, TextToSpeech, etc) y que sirva, a su vez, como soporte para el desarrollo de nuevos medios. Esta característica, sumada a la capacidad de simulación, posicionan al Squeak como una herramienta muy poderosa que permite una verdadera revolución en el manejo de la información. Sus características son:

- Squeak cuenta con las capacidades y herramientas típicas de un sistema Smalltalk y se encuentra entre los sistemas Smalltalk que se pueden ejecutar prácticamente en casi todas las plataformas de hardware. Su máquina virtual soporta compilación en tiempo de ejecución.

- Squeak puede ser descargado sin cargo alguno, junto con su código fuente. Existen actualmente debates sobre si la licencia de Squeak califica como software libre o no, debido a la presencia de una cláusula de indemnidad en la licencia original de Squeak. La versión 1.1 del sistema, originalmente aparecida en octubre de 1997 bajo la Squeak License, ha sido reemplazada en mayo del 2006 bajo la licencia de código abierto Apple Public Source License.

Alice

Alice es un lenguaje de programación funcional diseñado en el Programming Systems Lab de la Universidad del Sarre. Es un dialecto de Standard ML, extendido con soporte para evaluación perezosa, Concurrencia (Programación muti-hebras y programación distribuida por medio de llamada remota de

procedimientos) y Programación por restricciones.

La implementación de Alice de la Universidad Saarland utiliza la máquina virtual abstracta extensible SEAM (Simple Extensible Abstract Machine por sus siglas en inglés). Es un Software libre, que incluye compilación al vuelo hacia código binario propio así como para código de máquina para la arquitectura x86.

El mecanismo de llamada remota de procedimientos de Alice depende de su máquina virtual y permite realizar llamadas entre diferentes computadores.

Características:

- Al no existir en el entorno de programación Alice un editor del código de programación propiamente dich
- estudiantes que pueden programar en Alice se enfrentan con dificultades a la hora de introducirse en un lenguaje de programación tradicional que usa editor de texto (para la sintaxis).

mBlock



mBlock es el programa de programación por bloques de Makeblock para videojuegos y robots basado en Scratch 2.0.

mBlock 3 permite la programación de los robots fabricados por Makeblock hasta mediados de 2018: mBot, mBot Ranger además de otros productos compatibles de otras marcas basados en Arduino.

Este software permite comprobar en tiempo real como la programación por bloques se traduce a lenguaje C, así como añadir tus propias líneas de código al programa.

mBlock 5 es la nueva versión del software de programación por bloques de Makeblock. Basado en Scratch 3.0 cuenta con una apariencia más plana y limpia y añade funcionalidades y dispositivos extra a lo que ya podíamos hacer con su predecesor.

mBlock 5 es compatible con Arduino y los mBot como su predecesor, pero también con Codey Rocky y Neuron.

El editor alternativo en este caso es en Phyton en vez de C como su predecesor y se puede editar o comprobar en tiempo real.

mBlock 5 añade también funcionalidad IoT e IA.

Arduino IDE



La programación de Arduino es la programación de un microcontrolador. Esto era algo más de los ingenieros electrónicos, pero Arduino lo ha extendido a todo el público. Arduino ha socializado la tecnología.

Programar Arduino consiste en traducir a líneas de código las tareas automatizadas que queremos hacer leyendo de los sensores y en función de las condiciones del entorno programar la interacción con el mundo exterior mediante unos actuadores.

Arduino proporciona un entorno de programación sencillo y potente para programar, pero además incluye las herramientas necesarias para compilar el programa y “quemar” el programa ya compilado en la memoria flash del microcontrolador. Además el IDE nos ofrece un sistema de gestión de librerías y placas muy práctico. Como IDE es un software sencillo que carece de funciones avanzadas típicas de otros IDEs, pero suficiente para programar.

La plataforma Arduino se programa mediante el uso de un lenguaje propio basado en el lenguaje de programación de alto nivel Processing que es similar a C++. Arduino está basado en C y soporta todas las funciones del estándar C y algunas de C.

Android Studio



Android Studio es el entorno de desarrollo integrado oficial para la plataforma Android. Fue anunciado el

16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014. Está basado en el software IntelliJ IDEA de JetBrains y ha sido publicado de forma gratuita a través de la Licencia Apache 2.0. Está disponible para las plataformas Microsoft Windows, macOS y GNU/Linux. Ha sido diseñado específicamente para el desarrollo de Android.

Estuvo en etapa de vista previa de acceso temprano a partir de la versión 0.1, en mayo de 2013, y luego entró en etapa beta a partir de la versión 0.8, lanzada en junio de 2014. La primera compilación estable, la versión 1.0, fue lanzada en diciembre de 2014.

Se espera que se desarrollen nuevas funciones con cada versión de Android Studio. Las siguientes características se proporcionan en la versión estable actual:

- Integración de ProGuard y funciones de firma de aplicaciones.
- más Especificacion a la hora de Programar
- Renderizado en tiempo real
- Consola de desarrollador: consejos de optimización, ayuda para la traducción, estadísticas de uso.
- Soporte para construcción basada en Gradle.
- Refactorización específica de Android y arreglos rápidos.
- Un editor de diseño enriquecido que permite a los usuarios arrastrar y soltar componentes de la interfaz de usuario.
- Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versiones y otros problemas.
- Plantillas para crear diseños comunes de Android y otros componentes.
- Soporte para programar aplicaciones para Android Wear.

- Soporte integrado para Google Cloud Platform, que permite la integración con Google Cloud Messaging y App Engine.
- Un dispositivo virtual de Android que se utiliza para ejecutar y probar aplicaciones.

APP Inventor



App Inventor es un entorno de desarrollo de software creado por Google Labs para la elaboración de aplicaciones destinadas al sistema operativo Android. El usuario puede, de forma visual y a partir de un conjunto de herramientas básicas, ir enlazando una serie de bloques para crear la aplicación. El sistema es gratuito y se puede descargar fácilmente de la web. Las aplicaciones creadas con App Inventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil.

Con Google App Inventor, se espera un incremento importante en el número de aplicaciones para Android debido a dos grandes factores: la simplicidad de uso, que facilitará la aparición de un gran número de nuevas aplicaciones; y Google Play, el centro de distribución de aplicaciones para Android donde cualquier usuario puede distribuir sus creaciones libremente.

Características:

- ➢ Basado en [httpy Blockly] de JavaScript para crear un lenguaje visual. Estas librerías están distribuidas por Massachusetts Institute of Technology bajo su licencia libre.
- ➢ Permite crear una aplicación en menos tiempo que otros. y se pueden programar aplicaciones más complejas en mucho menos tiempo que con los lenguajes más tradicionales, basados en texto.
- ➢ La interfaz gráfica: permite al usuario crear aplicaciones con muchas funcionalidades.

SÍNTESIS CURRICULAR DE LOS AUTORES DEL LIBRO

RODRIGO GASTÓN MANRESA

Nació en la ciudad de Salta donde terminó sus estudios primarios, secundarios y terciarios en el Colegio N.° 8120 Santa Isabel de Hungría, en el Colegio N.° 5080 Doctor Manuel Antonio de Castro (Ex-Colegio Nacional) y en el Instituto Superior N.° 8063 María del Rosario de San Nicolás respectivamente, para luego emigrar a Buenos Aires donde egresó de la Universidad Tecnológica Nacional (UTN) con los títulos de Profesor en Disciplinas Industriales - Analista de Sistemas de Computación (2005) y de la Universidad Abierta Interamericana como Analista Programador (2002); posee tres pos-títulos, una Especialización Superior en Educación Tecnológica (2012), una Especialización Docente en Educación y TIC (2014) y otra Especialización Docente en Políticas y Programas Socioeducativos (2018). Actualmente se encuentra cursando la Maestría en Dirección de Tecnologías de la Información. Desde joven se vio

fascinado por las computadoras y durante su formación profesional realizó y asistió a numerosos cursos, seminarios y congresos relacionados al área de la tecnología informática y educativa. Está abocado a la docencia desde el año 2006, paso por varios establecimientos importantes de nivel secundario y actualmente se desempeña en varias carreras de nivel superior en cátedras relacionadas a la Informática y a la tecnología.

Es autor de varias publicaciones y ha obtenido varias menciones de honor y premios por su destacada labor en diversos proyectos. Es un entusiasta, activista y defensor del software libre, cualidades que lo llevaron a organizar y ser el coordinador de las sedes oficiales de su provincia en los mayores eventos de difusión de software libre: FLISoL (Festival Latinoamericano de Instalación de Software Libre) y SFD (Software Freedom Day) siendo estos declarados de interés por diversos organismos gubernamentales.

GONZALO JAVIER SUÁREZ

Nació en la ciudad de Salta donde realizó sus estudios primarios en la escuela Juan Bautista Alberdi y secundarios en la E.E.T. N.º 3137 Martina Silva de Gurruchaga, para luego realizar dos años en la Universidad Nacional de Salta en la carrera Licenciatura en Análisis de Sistemas, realizo un curso de informática básica orientada al mundo del trabajo en la EAP, en el año 2013 realizo un curso de robótica dictado por la fundación Dwengo, en el 2015 ingresa a estudiar en el nivel superior del Instituto General Manuel Belgrano N.º 6001 en la carrera Técnico Superior en Soporte y Mantenimiento Informático de la cual años más tarde se recibe.

MAURO ESTEBAN RODRÍGUEZ ZUBIETA

Estudió completando sus estudios de Analista Programador (1993) y Analista de Sistemas (1995), en el Instituto Facundo De Zuviria (Salta), trabajó en empresas como Plumada S.A. en el área de DATAWHEREHOUSING (2005-2006), Supermercados Luque (2010-2012) como encargado de taller de cómputos y entregando a la docencia de informática desde el año 1994 al 2010 con 16 años de experiencia DOCENTE en todos los niveles decidió retomar la programación y volver a sus origines.

En la actualidad esta involucrado en el proyecto OpenOffice y SuPort para Ecomstation/OS/2 y en el proyecto MONO.LINUX para la portación de las librerías .NET a entornos UNIX. Y desarrollándose como RTI (Responsable Técnico en Informática) en el programa CONECTAR IGUALDAD.

ROGELIO AUGUSTO SALGADO

Nació y vive en salta, actualmente cursa el último año de la carrera Tecnicatura Superior en Soporte y Mantenimiento Informático. En su nivel secundario se formó en una escuela técnica, empezando sus estudios superiores (durante cuatro años) no sabía realmente cuál era su meta, entre estudios indefinidos, trabajos, viajes e intentos de negocios; un gran amigo (motivador) viendo su situación le ayudó a redefinir sus objetivos. Donde decide dejar todo y empezar de nuevo. Finalizando en 2014 un curso intensivo de marketing digital, lo que le determinó enfocarse y estudiar más la informática. De ahí su pasión por las computadoras y el software libre lo llevo a conocer a su profesor de informática Gastón Manresa, el cual le dio la oportunidad de colaborar en un capítulo.

Agradece a las personas que le ayudaron en todo momento, amigos, familia, y a Dios.

ELIZABETH CAROLINA DÍAZ PEPENAL

Nació en la ciudad de Salta, y allí realizó los estudios primarios y secundarios. En el año 2009 obtuvo el titulo de Técnico Superior en Sistemas Informáticos con orientación en teleinformática. Por consiguiente en el 2012 alcanzó el título de Profesora en Sistemas Informáticos con orientación en teleinformática. Luego el 15 de diciembre de 2015 egresó con el titulo Técnico Superior en Mantenimiento y Soporte Informático.

Durante toda la formación profesional se capacitó realizando numerosos cursos relacionados al área. Desde el año 2009 hasta la actualidad trabaja en la Biblioteca Popular Pbro. Pablo Fernando Acuña de Villa Primavera realizando atención al público y en el 2015 trabajo en la escuela IN.P.RE.D como profesora de informática, con niños especiales.

EMILCE ISABEL RODÓ

Nació en la provincia de Salta. Allí realizó sus estudios primarios y secundarios. En el año 2015 egresa del Profesorado Superior de Lenguas Vivas N.° 6007 obteniendo el título como Profesora de Inglés. Posee una especialización en Políticas Socioeducativas (2018) y actualmente cursa la última materia, el trabajo de investigación, de la Licenciatura en Inglés en la Universidad Fasta.

En el año 2016 conoce al distinguido Profesor Manresa quien ha sido su influencia al mundo del software libre y por quien se ha incursionado en la investigación en materia de idioma y de tecnologías libres. La primera participación fue como colaboradora en el FLISoL del año 2016. Desde el año 2017 y hasta la actualidad ha participado también en calidad de disertante sobre la enseñanza de la lengua extranjera con software libre.

NATALIA DEL VALLE RODRÍGUEZ

Nació en el Dpto de Chicoana, Provincia de Salta. Realizó los estudios primarios en el Departamento de Cerrillos y años más tardes finalizaría los estudios secundarios en la ciudad de Salta Capital. En el año 2008 obtuvo el título de Profesora de Historia en la Universidad Nacional de Salta. A partir de allí ingresó en el mundo de la Docencia como profesora en diversas instituciones educativas tanto de nivel medio como Superior dependientes de los repartimientos públicos y privados. Actualmente trabaja como profesora de Historia Argentina y Latinoamérica en el Instituto de Educación Superior Gral. Manuel Belgrano Nº 6001 en la carrera de Profesorado para la Educación Primaria y en los colegios Secundarios Patrick Dragon School N.º 8086 y Juan Pablo II N.º 5056.

Realiza trabajo de investigación específicos de la disciplina: Historia y además participa de cursos referidos a las nuevas Tecnologías y la Educación.

PAULA BETIANA ARANCIBIA

Nacida en la provincia de Chubut, en Trelew. Curso sus estudios primarios en dicha ciudad, pero terminó su Séptimo Grado en la escuela Joaquín Castellanos N.º 007 de Villa Las Rosas, de la ciudad de Salta Capital. Estudios secundarios terminados en el Colegio Instituto Carlos Guido Spano también de la provincia de Salta. Actualmente cursando el 3er año de la carrera Profesorado de Educación Primaria en el Instituto de Educación Superior N.º 6001 de la provincia de Salta. Participó en la Feria de Ciencia y Tecnología instancia zonal (Esc. Dr. Benjamín Zorrilla), ORA-2019 y Feria de Proyectos Innovadores por el IITA (UCaSal), y disertó junto a sus docentes: Profesor Manresa y Profesora Sofía Juaréz, en el evento Software Freedom Day 2019 (UNSa). Trabaja actualmente en el Colegio San Rafaél Arcangel. A futuro seguirá formándose en el ámbito educativo.

JORGE LUIS VENTURA

Nació en la ciudad de Salta donde cursó sus estudios primarios y secundarios completos en la Escuela N.º 4045 “Dr. Arturo Oñativia” y el Colegio N.° 5024 “Sargento Cabral” (2005). Estudio Técnico en reparación de PC en los institutos “INI Computación” e “IAC”. Ingreso a la Universidad Nacional de Salta, actualmente cursa la carrera de “Profesorado en Ciencias Biológicas”. Posteriormente ingreso al Instituto de Formación Docente 6001 “General Manuel Belgrano” (2012), donde cursa actualmente la carrera “Profesorado en Educación Primaria”. Trabaja de forma independiente como técnico en reparación y mantenimiento de PC’s.

MARCOS ALEJANDRO LÓPEZ PEREYRA

Nació en la ciudad de Salta Argentina, donde finalizo sus estudios primarios y secundarios. En el año 2008 inicio sus primeros pasos en el mundo de la informática y Electrónica Gral. desarrollando cursos básicos y avanzados de mantenimiento y reparación de computadoras y teléfonos celulares, electrodomésticos, diseño de proyectos electrónicos y diseño gráfico avanzado. En el 2012 curso un año en la carrera tecnicatura en electrónica de la Universidad Nacional de Salta y posteriormente la carrera “tecnicatura superior en mantenimiento y soporte informático”. Es aficionado al software libre dedicado a Linux y Android. Actualmente trabaja como profesor en reparación de computadoras y electrónica en empresa privada dedicada al tema e independientemente en laboratorio de diagnóstico avanzado y reparación de dispositivos electrónicos y microelectrónica.

MIGUEL ÁNGEL GILIBERTI

Vive en Salta, Argentina. Finalizó sus estudios secundarios en el colegio N.º 5095 Gral. Manuel Belgrano. Años más tarde, luego de haber cursado por un tiempo la carrera de Analista de Sistemas en la Universidad Nacional de Salta, regresa al edificio que alguna vez fue su colegio secundario de día y que cuando cae la noche, alberga la institución que ofrece la carrera que actualmente está realizando, la de Técnico Superior en Soporte y Mantenimiento Informático.

Desde temprana edad mostró determinado interés por ciertos aspectos de la informática y creció aprendiendo acerca de la misma de manera autodidacta. Mientras se encontraba realizando sus estudios secundarios, realizó un curso de Reparación de PC que ofrecía un instituto de la ciudad.

Con la llegada de Conectar Igualdad y sus netbooks, se convirtió en un usuario casual del

sistema operativo GNU/Linux. Mucho tiempo después eso cambió gradualmente cuando, con orientación por parte del Profesor Rodrigo Gastón Manresa, empezó a adentrarse y a participar en la comunidad y en los eventos de software libre que se realizan en Salta.

Anteriormente también formó (y forma) parte de comunidades en redes sociales tales como ITL (I.T Legion) en las que, junto a otros miembros, llevó a cabo programas de radio (FM y Online) dedicados a lo que es el sector de la tecnología.

ÁNGEL YAEL EDUARDO MOYA

Nació en la ciudad de Salta capital, Argentina, finalizó sus estudios primarios en la Esc. Dr. Benjamín Zorrilla N.° 4022 (2012), finalizó sus estudios secundarios en el Colegio Polimodal N.º 5159 (2017) siendo autodidacta en temas de informática, donde en cuarto año conoció a su profesora **Liliana Farfan** quien se convertiría en ese entonces en su mayor influencia y guía en el amplio mundo de la informática y programación brindándole oportunidades de participar en Soluciones para el Futuro By Samsung (2017) y el proyecto **programando para jugar** en la localidad de Molinos y Salta Capital (2016-2017).

En 2019 comenzó sus estudios de grado en la carrera Tecnicatura Superior en Soporte y Mantenimiento Informático, conociendo en primer año a su profesor **Rodrigo Gastón Manresa** quien se convirtió en su nueva referencia y gran guía. En

ese mismo año (2019) llevaron a cabo, con la destacada participación de su profesor, su proyecto **FREEGRAMACIÓN (programación de software libre al servicio de la educación)** presentándose en la **Feria de Ciencias instancia zonal** (**Esc. Dr. Benjamín Zorrilla**), **Feria de Ciencias instancia provincial** (**CCS**), **ORA-2019** por el **IITA** (**UcaSa**l), **Innova Técnicas Salta 2019** (**CCS**). Ganó el **concurso de arte y diseño digital con software libre** en el evento **Software Freedom Day 2019** (**UNSa**). En la actualidad es un apasionado, activista por el software libre y las nuevas tecnologías, agradeciendo a familiares, profesores y amistades quienes lo ayudaron a convertirse en todo lo que es hoy, apuntando a continuar la carrera Licenciatura en análisis de sistemas en la Universidad nacional de Salta. “**La vida siempre da revancha.”**

“compartir el conocimiento es generar riqueza intelectual”



Hacia la Libertad Tecnológica

Epílogo:

Vivimos en una sociedad donde la tecnología ocupa un lugar en nuestro quehacer cotidiano, laboral, entre otros, es por ello que se nos abren varias posibilidades al conectarse a una red, conocer la filosofía que inunda el mundo del software libre permitirá al usuario a elegir mejor los programas.

Observarán la cultura tecnológica en la cual todos participamos y a su vez en este libro conocerán la importancia y aplicación de esta para nuestras actividades.